

*Modeliranje enostavnih bioloških preklopnih sistemov z
uporabo mehke logike in razširjenih Petrijevih mrež*

Jure Bordon

DOKTORSKA DISERTACIJA

PREDANA

FAKULTETI ZA RAČUNALNIŠTVO IN INFORMATIKO

KOT DEL IZPOLNJEVANJA POGOJEV ZA PRIDOBITEV NAZIVA

DOKTOR ZNANOSTI

S PODROČJA

RAČUNALNIŠTVA IN INFORMATIKE



Ljubljana, 2017

*Modeliranje enostavnih bioloških preklopnih sistemov z
uporabo mehke logike in razširjenih Petrijevih mrež*

Jure Bordon

DOKTORSKA DISERTACIJA

PREDANA

FAKULTETI ZA RAČUNALNIŠTVO IN INFORMATIKO

KOT DEL IZPOLNJEVANJA POGOJEV ZA PRIDOBITEV NAZIVA

DOKTOR ZNANOSTI

S PODROČJA

RAČUNALNIŠTVA IN INFORMATIKE



Ljubljana, 2017

IZJAVA

Izjavljam, da sem avtor dela in da slednje ne vsebuje materiala, ki bi ga kdorkoli predhodno že objavil ali oddal v obravnavo za pridobitev naziva na univerzi ali na drugem visokošolskem zavodu, razen v primerih, kjer so navedeni viri.

— Jure Bordon —
februar 2017

ODDAJO SO ODOBRILI

dr. Nikolaj Zimic

redni profesor, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani

PREDSEDNIK OCENJEVALNE KOMISIJE

dr. Branko Šter

redni profesor, Fakulteta za računalništvo in informatiko, Univerza v Ljubljani

ČLAN OCENJEVALNE KOMISIJE

dr. Aleš Belič

redni profesor, Lek farmacevtska družba d.d.

ZUNANJI ČLAN OCENJEVALNE KOMISIJE

dr. Roman Jerala

redni profesor, Kemijski inštitut Ljubljana, Slovenija

ZUNANJI ČLAN OCENJEVALNE KOMISIJE

PREDHODNA OBJAVA

Izjavljam, da so bili rezultati obravnavane raziskave predhodno objavljeni/sprejeti za objavo v recenzirani reviji ali javno predstavljeni v naslednjih primerih:

- [1] J. Bordon, M. Moškon, N. Zimic and M. Mraz Fuzzy logic as a computational tool for quantitative modelling of biological systems with uncertain kinetic data. *IEEE/ACM transactions on computational biology and bioinformatics*, 12 (5), 1199-1205, 2015. doi: [10.1109/TCBB.2015.2424424](https://doi.org/10.1109/TCBB.2015.2424424)

Potrjujem, da sem pridobil pisna dovoljenja vseh lastnikov avtorskih pravic, ki mi dovoljujejo vključitev zgoraj navedenega materiala v pričujočo disertacijo. Potrjujem, da zgoraj navedeni material opisuje rezultate raziskav, izvedenih v času mojega podiplomskega študija na Univerzi v Ljubljani.

"All models are wrong, but some are useful". –George E. P. Box

POVZETEK

Modeliranje in simulacija sta v zadnjih dveh desetletjih postala nepogrešljiva za načrtovanje novih in analizo obstoječih bioloških sistemov. Z uporabo njunih različnih pristopov lahko občutno znižamo čas in stroške načrtovanja ter izvajanja laboratorijskih poizkusov. Izbira metode modeliranja je odvisna od željene natančnosti simulacijskih rezultatov in podatkov o sistemu, ki jih imamo na voljo. Kvalitativni pristopi se uporabljajo za opis osnovnih dinamičnih lastnosti in topologije gensko regulatornega omrežja ter služijo kot izhodišče za bolj natančne pristope. Po drugi plati kvantitativni pristopi dinamiko sistema opišejo zelo natančno, a za izračun simulacijskih rezultatov potrebujejo natančne vrednosti kinetičnih parametrov, ki pogosto niso na voljo, njihovo pridobivanje pa je dolgotrajno in posledično drago. Kot močno grafično in matematično modelirno orodje so se tako za kvalitativne kot tudi kvantitativne pristope uveljavile Petrijeve mreže. V pričujoči doktorski disertaciji predstavimo kvantitativno metodo za modeliranje, ki temelji na mehki logiki in Petrijevih mrežah. Model procesa, ki ga vzpostavimo na podlagi opisnega izkustvenega poznavanja, za pridobitev kvantitativno relevantnih simulacijskih rezultatov ne potrebuje natančnih vrednosti kinetičnih parametrov, ki pogojujejo dinamiko sistema. Definiciji zveznih Petrijevih mrež dodamo mehke prožilne funkcije, ki jih lahko uporabimo za modeliranje procesov z manjkajočimi vrednostmi kinetičnih parametrov. Predlagan pristop se lahko uporabi tudi kot dopolnitev obstoječih metod za kvantitativno modeliranje v primeru, da so vrednosti kinetičnih parametrov sistema večinoma poznane. Metodo, ki temelji na mehki logiki, uporabimo za izgradnjo modela hipotetičnega represilatorja s tremi členi, z mehкими zveznimi Petrijevimi mrežami pa zgradimo model cirkadianega ritma glive *Neurospora*. Pokažemo, da simulacijski rezultati modela, ki vsebuje mehke opise posameznih procesov, ohranijo kvantitativno biološko relevantnost simulacijskih rezultatov v primerjavi s simulacijskimi rezultati modela, ki je v celoti zgrajen na podlagi

obstojećih metod, ki za vhode uporabljajo znane vrednosti kinetičnih parametrov.

Ključne besede: modeliranje bioloških sistemov, neznani kinetični parametri, mehka logika, Petrijeve mreže

ABSTRACT

Modelling and simulation have become indispensable in the design of novel and analysis of existing biological systems. Using various approaches, they can significantly reduce the time and cost of planning and executing laboratory experiments. The choice of modelling technique depends on the desired accuracy of simulation results and the available kinetic data of the system. Qualitative approaches can be used to describe basic dynamical properties and topology of the underlying network. On the other hand, quantitative approaches are used for detailed modeling of system's dynamics. However, exact kinetic data that are required to obtain simulation results using state-of-the-art quantitative methods are often missing and are hard or even impossible to obtain. Both qualitative and quantitative approaches can be graphically represented as Petri nets, which serve as a powerful framework for constructing a biological system model. In this dissertation we present a quantitative fuzzy logic modelling approach that is able to cope with unknown kinetic data by using expert knowledge to model process descriptions and can thus produce quantitatively relevant simulation results even when kinetic data are incomplete or only vaguely defined. In addition, we extend the continuous Petri net definition and introduce fuzzy firing rate functions. Moreover, the approach can be used in the combination with the existing quantitative modelling techniques only in certain parts of the system, i.e. where kinetic data are missing. We use the fuzzy logic based approach to construct a model of a hypothetical three-gene repressilator and fuzzy continuous Petri nets for constructing a *Neurospora* circadian rhythm model. Simulation results obtained with fuzzy approach show that using fuzzy logic for describing processes with unknown kinetic does not significantly affect the quantitative aspects of the model.

Key words: modelling biological systems, unknown kinetic rates, fuzzy logic, Petri nets

ZAHVALA

Na prvem mestu se zahvaljujem mentorju prof. dr. Mihi Mrazu za usmerjanje, komentiranje in spodbudo. In za potrpežljivost.

Hvala Mihi Moškonu za konstruktivne debate in dodatno pomoč.

Zahvaljujem se tudi celotnemu kolektivu LRSS za premore med delom in popestritev vzdušja v težjih trenutkih ter članom LASPP za produktivne in malo manj produktivne ure, ki so pripomogle, da je delo običajno steklo še pred koncem dneva.

Hvala atu, mami in bratu z družino, ker so prenašali mojo odsotnost in me niso prepogosto spraševali kdaj bom zaključil z delom.

Nenazadnje hvala Špeli in Timu za veliko več kot samo njune nasmeške in dobro voljo.

— Jure Bordon, Ljubljana, februar 2017.

KAZALO

<i>Povzetek</i>	<i>i</i>
<i>Abstract</i>	<i>iii</i>
<i>Zahvala</i>	<i>v</i>
1 <i>Uvod</i>	1
1.1 Znanstveni prispevki disertacije	3
1.2 Metodologije dela	4
1.3 Pregled realizacije disertacije	5
2 <i>Osnovni pojmi računske biologije in gensko regulatornih omrežij</i>	7
2.1 Sintezna, sistemska in računska biologija	8
2.2 Biološki sistemi	8
2.2.1 Kemijske zvrsti in procesi v biološkem sistemu	8
2.2.2 Gensko regulatorna omrežja	9
2.3 Dinamika v bioloških sistemih	10
3 <i>Modeliranje dinamike enostavnih bioloških sistemov</i>	13
3.1 Kvalitativno modeliranje	14
3.2 Kvantitativno modeliranje	15
3.2.1 Deterministični pristopi	16
3.2.2 Stohastični pristopi	21
3.3 Omejitve obstojecih pristopov	22

4	<i>Osnove mehke logike</i>	25
4.1	Mehka množica	26
4.2	Mehka spremenljivka	27
4.3	Mehki odločitveni sistem	29
4.4	Vzorčni primer mehkega odločitvenega sistema	30
4.4.1	Določanje mehkih spremenljivk	31
4.4.2	Mehko odločanje	33
4.4.3	Ostrenje vrednosti mehkih spremenljivk	37
5	<i>Uporaba mehke logike za modeliranje dinamike bioloških sistemov</i>	39
5.1	Pregled literature	41
5.1.1	Izgradnja pol-kvantitativnega modela dinamike biološkega procesa na podlagi mehke logike in približnega izkustvenega znanja	42
5.1.2	Izgradnja kvantitativnega mehkega modela dinamike biološkega procesa na podlagi učenja iz eksperimentalnih podatkov	45
5.1.3	Odkrivanje enostavnih relacij v topologiji gensko regulatornega omrežja na podlagi eksperimentalnih podatkov	46
5.1.4	Odkrivanje kompleksnih relacij v topologiji gensko regulatornega omrežja na podlagi eksperimentalnih podatkov	47
5.2	Mehki pol-kvantitativni model Higgins-Sel'kovskega oscilatorja	48
6	<i>Modeliranje dinamike enostavnih bioloških procesov z neznanimi vrednostmi kinetičnih parametrov</i>	53
6.1	Model genetskega represilatorja	54
6.2	Referenčni deterministični model represilatorja s tremi členi	55
6.3	Postopek izgradnje mehkega modela biološkega procesa z manjkajočimi vrednostmi parametrov	56
6.4	Izgradnja mehkega modela represije transkripcije	58
6.4.1	Identifikacija spremenljivk ter razpoložljivih in manjkajočih parametrov opazovanega procesa	58
6.4.2	Analiza odvisnosti procesa transkripcije od vrednosti parametrov	59
6.4.3	Identifikacija lingvističnega opisa dinamike procesa	59
6.4.4	Izgradnja mehkega modela	61
6.5	Združevanje mehkega modela in modela na podlagi diferencialnih enačb	64

6.6	Rezultati	67
6.7	Vrednotenje metode in diskusija	69
6.7.1	Modeli in metrike za vrednotenje natančnosti mehkega modela	70
6.7.2	Odzivi modelov pri različnih vrednostih vhodne spremenljivke	72
6.7.3	Časovni odzivi modelov pri invertiranem pulzu vrednosti vhodne spremenljivke	73
6.7.4	Primerjava simulacijskih rezultatov	74
6.7.5	Diskusija	75
7	<i>Vpeljava mehke logike v zvezne Petrijeve mreže</i>	77
7.1	Petrijeve mreže	78
7.1.1	Osnovna definicija Petrijevih mrež	78
7.1.2	Graf Petrijeve mreže	80
7.1.3	Označitev Petrijeve mreže	81
7.2	Mehke Petrijeve mreže	82
7.3	Zvezne Petrijeve mreže	85
7.4	Uporaba Petrijevih mrež za modeliranje bioloških sistemov	86
7.5	Pomankljivosti obstoječih definicij Petrijevih mrež	87
7.6	Zvezne Petrijeve mreže z mehko logiko	88
7.7	Modeliranje bioloških procesov z zveznimi Petrijevimi mrežami z mehko logiko	91
7.7.1	Mehka prožilna funkcija za modeliranje degradacije	92
7.7.2	Mehka prožilna funkcija za modeliranje translacije	93
7.7.3	Mehka prožilna funkcija za modeliranje transkripcije	94
7.8	Zvezna Petrijeva mreža z mehko logiko za modeliranje cirkadianega ritma glive <i>Neurospora</i>	97
7.8.1	Referenčni deterministični model glive <i>Neurospora</i>	97
7.8.2	Model z mehкими prožilnimi funkcijami	99
7.8.3	Uporaba mehkih prožilnih funkcij za modeliranje bioloških procesov glive <i>Neurospora</i>	102
8	<i>Zaključek</i>	107
8.1	Glavna prispevka k znanosti	108
8.2	Nadaljnje delo	109

Uvod

Napredek na raziskovalnih področjih sintezne in sistemske biologije v zadnjih dveh desetletjih je omogočil načrtovanje in izgradnjo novih ter razumevanje delovanja obstoječih bioloških sistemov [1–3]. Izgradnja novega biološkega sistema temelji na načrtnem spreminjanju posameznega biološkega procesa, npr. pohitritvi ali upočasnitvi procesa, oziroma na realizaciji novega gensko regulatornega omrežja (angl. *gene regulatory network* - GRN). Med pomembne biološke sisteme spadajo različna genetska stikala (angl. *genetic toggle-switch*) [4] in genetski ter metabolni oscilatorji (angl. *genetic/metabolic oscillator*) [5], ki opravljajo ključne funkcije v številnih organizmih, kot so odziv na zunanji impulz, periodično spreminjanje stanja, cirkadiani ritem itd. Pojavljajo se v različnih oblikah, npr. v obliki bistabilnega stikala za gensko terapijo, represilatorja, metabolnega oscilatorja itd., najpogosteje pa jih zasnujemo kot nova gensko regulatorna omrežja (GRO) [4–8]. Ker sta zasnova ter *in vivo* implementacija časovno in cenovno potratna procesa, lahko z različnimi načini modeliranja načrtovanje in realizacijo bioloških sistemov načrtovanih na podlagi GRO pohitrismo in pocenimo [9, 10].

Izbira metode modeliranja je odvisna od velikosti biološkega sistema, čigar dinamiko želimo opisati, želene natančnosti simulacijskih rezultatov in od tega, ali kinetične parametre, ki določajo dinamične zakonitosti sistema, poznamo dovolj natančno. *Kvalitativne metode* modeliranja nam omogočajo osnovni opis delovanja sistema in procesov (npr. proces je aktiven, v sistemu procesu A sledi proces B itd.), pri čemer časa in koncentracije opazovanih kemijskih zvrsti ne obravnavamo kvantitativno in te posledično nimajo merskih enot. Za natančen opis dinamike biološkega sistema, pri katerem opazujemo, kako se koncentracije opazovanih kemijskih zvrsti spreminjajo v odvisnosti od časa, uporabljamo *kvantitativne metode* modeliranja. Slednje večinoma temeljijo na glavni kemijski enačbi (angl. *chemical master equation* - CME) (stohastično modeliranje) ali na numeričnem reševanju sistemov navadnih diferencialnih enačb (angl. *ordinary differential equations* - ODE) (deterministično modeliranje). Kvantitativni modeli dinamiko sistema sicer opišejo bolj natančno [11–14], a za izračun simulacijskih rezultatov potrebujejo natančne vrednosti širokega nabora kinetičnih parametrov (angl. *kinetic parameters*). V bioloških sistemih nastopa več osnovnih bioloških procesov (transkripcija, translacija, degradacija itd.), delovanje vsakega od njih pa lahko opišemo z eno ali več kemijskimi reakcijami (vezava, cepitev, dimerizacija itd.). V vsaki izmed slednjih nastopa vsaj en kinetičen parameter, ki določa hitrost reakcije. Število kinetičnih parametrov z večanjem sistema zato narašča zelo hitro, vrednosti teh

parametrov pa pogosto niso znane. Z metodami za ocenjevanje vrednosti kinetičnih parametrov lahko slednje ocenimo na podlagi eksperimentalnih podatkov, a je njihov zajem zaradi omejitev pri meritvah pogosto neizvedljiv [15].

V zadnjih letih se kot alternativna metoda za modeliranje bioloških sistemov uveljavlja metodologija mehke logike [16], ki stanje in dinamiko sistema opisuje na lingvistični način (npr. *ob visoki koncentraciji proteina A je izražanje gena utišano*). Obstoječi pristopi, ki temeljijo na mehki logiki in ne uporabljajo kinetičnih parametrov, ker vrednosti slednjih niso znane, dinamiko sistema opisujejo zgolj kvalitativno [17] ali opisujejo le posamične dogodke v sistemu (npr. ali je izražanje gena aktivirano ali utišano, ali se bo vezava transkripcijskega faktorja zgodila itd.) [18, 19], ne opisujejo pa sprememb absolutnih koncentracij kemijskih vrst v sistemu. V primeru, da so vrednosti kinetičnih parametrov znane, je natančnost kvantitativnih pristopov, ki temeljijo na uporabi metod mehke logike, primerljiva z natančnostjo ostalih determinističnih metod [20].

Za ponazoritev modelov enostavnih bioloških sistemov v grafični obliki, ki predstavlja omrežje interakcij, so se kot močno modelirno orodje uveljavile Petrijeve mreže. Za potrebe modeliranja bioloških sistemov se uporablja različne razširjene definicije Petrijevih mrež, ki omogočajo izgradnjo tako kvalitativnih kot tudi kvantitativnih modelov [21]. Obstoječe metode za kvantitativno modeliranje, podobno kot razširjene Petrijeve mreže, za pridobitev simulacijskih rezultatov dinamike enostavnih bioloških sistemov potrebujejo natančne vrednosti kinetičnih parametrov. V primeru, da natančne vrednosti kinetičnih parametrov niso znane in jih zaradi manjkajočih eksperimentalnih podatkov ni mogoče oceniti, z obstoječimi kvantitativnimi metodami ne moremo pridobiti simulacijskih rezultatov [22].

1.1 Znanstveni prispevki disertacije

V disertaciji predstavimo naslednja prispevka k znanosti:

- *Vzpostavitev nove metode za modeliranje bioloških sistemov*: Vzpostavljena je nova metoda za modeliranje bioloških sistemov, ki temelji na mehki logiki in Petrijevih mrežah. Metoda je primerna za modeliranje bioloških sistemov, pri katerih natančnih vrednosti kinetičnih parametrov, ki določajo dinamične zakonitosti sistema, ne poznamo. Za izračun simulacijskih rezultatov natančnih vrednosti kinetičnih parametrov ne potrebujemo, natančnost metode pa je odvisna od našega izkusnega poznavanja delovanja posameznih gradnikov sistema. Upo-

rablja se lahko za vzpostavitev modela celotnega biološkega sistema ali kot dopolnitev modela pri tistih delih sistema, kjer natančne vrednosti kinetičnih parametrov niso znane, zaradi česar uporaba obstoječih metod ni možna. Vzpostavitev nove metode je opisana v poglavju 6.

- *Razširitev definicije Petrijevih mrež:* Razširjena je obstoječa definicija Petrijevih mrež, kar omogoča vpeljavo uporabe metodologije mehke logike in poveča njihovo uporabnost pri modeliranju bioloških sistemov. S tem uporabo razširjenih Petrijevih mrež razširimo na pol-kvantitativne modele, ki se jih uporablja za modeliranje bioloških sistemov, za katere natančne vrednosti kinetičnih parametrov niso znane. Razširjena definicija Petrijevih mrež je opisana v poglavju 7.

1.2 Metodologije dela

Za raziskovalno delo, predstavljeno v pričujočem doktorskem delu, so bile uporabljene naslednje metode:

- Pregled in analiza literature: Narejen je bil pregled literature modeliranja v sintezni in sistemski biologiji, pregled uporabe determinističnih in stohastičnih metod modeliranja, metode mehke logike in uporabe mehke logike v Petrijevih mrežah za modeliranje bioloških sistemov.
- Vzpostavitev nove metode modeliranja z mehko logiko: Vzpostavljena je bila metoda za modeliranje bioloških procesov z neznanimi vrednostmi kinetičnih parametrov, ki pogojujejo dinamiko sistema. Metoda temelji na mehki logiki, pri kateri je za izgradnjo modela uporabljeno izkustveno opisno poznavanje procesa.
- Razširitev definicije zveznih Petrijevih mrež: Definirana je bila zvezna Petrijeva mreža z mehko logiko, ki poleg navadnih vsebuje tudi mehke prožilne funkcije. Slednje nam pri simulaciji s Petrijevim mrežami omogočajo uporabo mehke logike. Mehke prožilne funkcije delujejo na osnovi vzpostavljene metode iz predhodne alineje.
- Implementacija modelov: V programskem okolju MATLAB Simulink® sta bila z uporabo obstoječih determinističnih metod ter z vzpostavljeno metodo mehke logike implementirana modela dveh bioloških sistemov, in sicer hipotetičnega

represilatorja s tremi členi in cirkadianega ritma modelnega organizma glive *Neurospora*.

- Analiza in primerjava rezultatov: Analizirani in primerjani so bili rezultati obstoječih metod in vzpostavljene metode vseh implementiranih modelov za vsakega od bioloških sistemov.

1.3 Pregled realizacije disertacije

V pričujoči doktorski disertaciji predstavimo novo metodo za modeliranje dinamike enostavnih bioloških sistemov, ki temelji na uporabi metode mehke logike in opisnem izkustvenem poznavanju zakonitosti dinamike bioloških procesov. V poglavju 2 predstavimo osnovne pojme sintezne in systemske biologije, ki jih v nalogi uporabljamo kot izhodišče za vzpostavitev modelov bioloških sistemov. V poglavju 3 naredimo pregled literature s področja modeliranja dinamike enostavnih bioloških sistemov. Predstavimo tako kvalitativne kot tudi kvantitativne pristope ter izpostavimo pomanjkljivosti obeh skupin. V poglavju 4 predstavimo osnove mehke logike in razložimo pojme, ki se nanašajo na uporabo mehke logike za modeliranje, kot so mehka spremenljivka, mehko odločanje itd., ter na primeru mehkega krmilnika za nadzor ambientalne temperature prikažemo osnove delovanje mehkega odločitvenega sistema. Pregled primerov uporabe mehke logike za modeliranje enostavnih bioloških sistemov podrobneje predstavimo v poglavju 5. Izpostavimo najpomembnejše pristope za modeliranje dinamike enostavnih bioloških sistemov in na primeru uporabe enostavnega sintetičnega biološkega oscilatorja prikažemo njihove pomanjkljivosti. Metodo, ki temelji na podlagi mehke logike in ki omenjene pomanjkljivosti odpravlja, vzpostavimo v poglavju 6. Rezultate vzpostavljene metode prikažemo na primeru hipotetičnega represilatorja s tremi členi. V poglavju 7 mehko logiko vpeljemo v definicijo zveznih Petrijevih mrež. Zvezne Petrijeve mreže z mehko logiko nato uporabimo za izgradnjo modela cirkadianega ritma vzorčnega organizma glive *Neurospora*. Simulacijske rezultate modelov, zgrajenih z zveznimi Petrijevim mrežami z mehko logiko, primerjamo s simulacijskimi rezultati modela, zgrajenega na podlagi obstoječih metod. V poglavju 8 sledijo zaključek in izhodišča za nadaljnje delo.



*Osnovni pojmi računske
biologije in gensko
regulatornih omrežij*

V pričujočem poglavju predstavimo osnovne pojme *sinetzne* in *sistemske biologije*, ki sta predmet obravnave v pričujoči doktorski disertaciji, a niso neposredno povezani s področjem računalništva.

2.1 *Sinetzna, sistemska in računska biologija*

Sinetzna biologija je interdisciplinarna veda, ki združuje znanja biologije in biotehnologije ter inženirskih principov. Namen sinetzne biologije je uporaba znanj za spreminjanje obstoječih in načrtovanje ter realizacijo novih bioloških sistemov [23]. *Sistemska biologija* je veda, katere predmet je holistično razumevanje bioloških procesov, celic in organizmov [24]. *Računska biologija* je raziskovalno področje podatkovnega rudarjenja, računalniškega modeliranja in simulacije za potrebe razumevanja dinamike sistemov na področju ved o življenju [25].

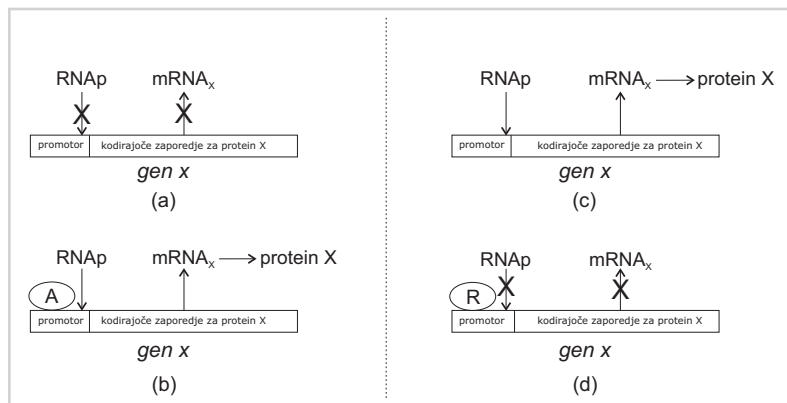
2.2 *Biološki sistemi*

S pojmom *biološki sistemi* opišemo veliko vrst sistemov, ki jih obravnavata sistemska in sinetzna biologija in se razlikujejo v kompleksnosti in velikosti, npr. bakterije, večcelični organizmi, organi itd. Kot biološke sisteme v pričujoči disertaciji smatramo poenostavljene celične podsisteme, v katerih opazujemo procese na molekularni ravni. Osnovne entitete, ki jih v sistemu opazujemo, imenujemo *kemijske zvrsti*. Slednje predstavljajo več tipov molekul, ki se razlikujejo po svoji velikosti, sestavi in vlogah v sistemu, npr. *gen*, *mRNA*, *protein*. Pri modeliranju dinamike enostavnih bioloških sistemov opisujemo interakcije med posameznimi kemijskimi zvrstmi in vpliv interakcij na spremembo stanja sistema.

2.2.1 *Kemijske zvrsti in procesi v biološkem sistemu*

Geni so del DNA (angl. *deoxyribonucleic acid* - DNA) in so sestavljeni iz *promotorskega dela*, ki sodeluje pri uravnavanju izražanja gena, in *kodirajočega dela*, ki predstavlja navodilo za izgradnjo ciljnega proteina. Na promotorski del gena se veže encim *RNA polimeraza*, kar sproži prepis oziroma proces *transkripcije* kodirajočega dela gena v molekulo mRNA (angl. *messenger RNA*). Vezava RNA polimeraze je encimska reakcija, pri kateri se *substrat* (DNA) veže z encimom (RNA polimeraza) in tvori *kompleks* (DNA z vezano RNA polimerazo), s pomočjo katerega nato nastaja *produkt* (mRNA). Molekulo mRNA, ki je rezultat transkripcije, *ribosom* prevede v aminokislinsko zaporedje, ki

se praviloma zvije v ciljni protein. Proces prevajanja mRNA v pripadajoče aminokislinsko zaporedje imenujemo *translacija*. Proteini, ki nastanejo kot posledica transkripcije in translacije, lahko v sistemu nastopajo v vlogi *transkripcijskih faktorjev*. Slednji se vežejo na promotor enega od genov in s tem omogočijo oziroma pospešujejo vezavo RNA polimeraze (aktivatorji) ali vezavo onemogočijo oziroma jo zavirajo (represorji). Shemo aktivacije in represije izražanja gena prikazuje slika 2.1.



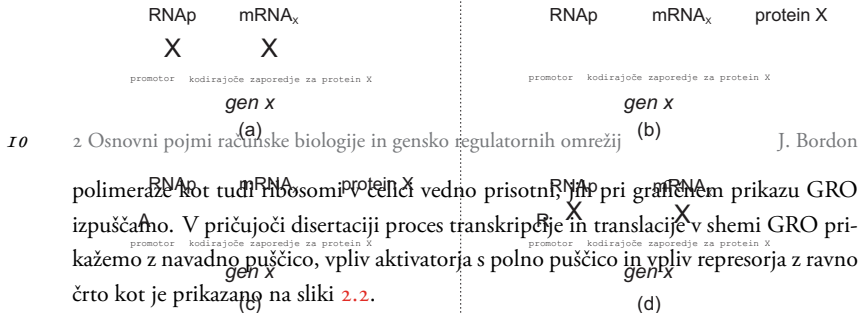
Slika 2.1

Proces aktivacije (levo) in represije (desno) izražanja gena. Odsotnost aktivatorja *A* onemogoči transkripcijo (a), medtem ko njegova prisotnost vezavo RNA polimeraze RNAP, ki sproži proces transkripcije, omogoči oziroma pohiti (b). Odsotnost represorja vezavo RNA polimerazi ne preprečuje, kar pomeni, da proces transkripcije poteka nemoteno (c), medtem ko prisotnost represorja *R* vezavo RNA polimeraze zavira, zato se proces transkripcije upočasni oziroma ustavi (d).

Pri modeliranju genske ekspresije v bioloških sistemih običajno predpostavljamo, da je RNA polimeraza v celici vedno na voljo in da na proces transkripcije vplivata zgolj prisotnost ali odsotnost transkripcijskih faktorjev. Protein v vlogi transkripcijskega faktorja vpliva na izražanje genov v sistemu, pri čemer lahko vpliva tudi na izražanje lastnega gena. V vseh bioloških sistemih se odvija tudi proces *razgradnje* ali *degradacije*, zaradi katerega se količina posamezne kemijske zvrsti v času zmanjšuje. Gene in produkte njihovega izražanja na osnovi omenjenih procesov povezujemo v *gensko regulatorna omrežja*.

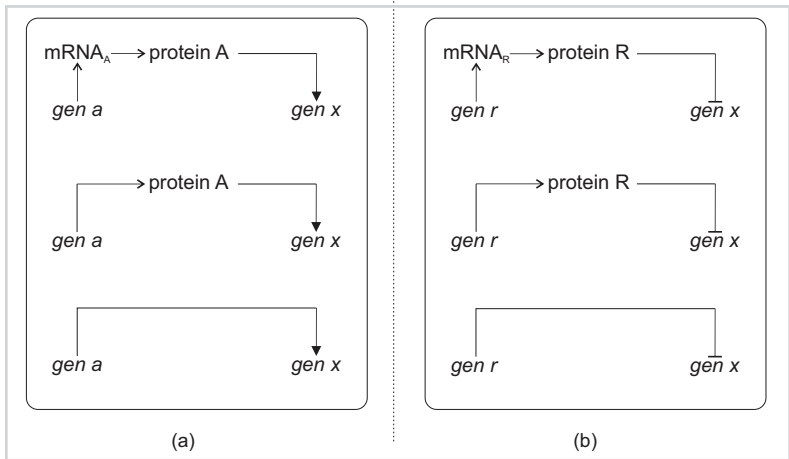
2.2.2 Gensko regulatorna omrežja

Gensko regulatorna omrežja (GRO) so množice genov, katerih produkti vplivajo na izražanje vsaj enega od opazovanih genov [26]. GRO lahko predstavimo z grafom, v katerem kemijske zvrsti nastopajo kot vozlišča, povezave pa predstavljajo procese degradacije, translacije ali transkripcije. Ker predpostavljamo, da so tako encimi RNA



Slika 2.2

Različni načini označitve aktivacije (a) in represije (b) v GRO. V primeru zgornjih delov leve in desne strani sheme opazujemo procese transkripcije, translacije in vpliv produkta na enega od preostalih genov, pri sredinskih primerih proces transkripcije izpustimo, najbolj poenostavljena pa sta spodnja primera, pri katerih povežemo le gena, s čimer želimo prikazati vpliv produkta gena *a* oziroma *r* na izražanje gena *x*.



V določenih primerih vmesne procese (transkripcijo in/ali translacijo) zaradi poenostavitve prikaza omrežja iz sheme izpustimo in s povezavo, ki prikazuje vpliv transkripcijskega faktorja, povežemo le gena. Na podlagi sheme zgradimo model, pri čemer je cilj modeliranja biološkega sistema na podlagi GRO opazovanje njegove *dinamike*.

2.3 Dinamika v bioloških sistemih

Dinamika bioloških sistemov opisuje spremembe *koncentracij* vseh opazovanih kemijskih zvrsti v odvisnosti od časa. Spremembe koncentracij zapišemo z izrazoma

$$\mathbf{x}(t) = \{x_1, x_2, \dots, x_N\}, \quad (2.1)$$

$$\mathbf{x}(t + dt) = \{x'_1, x'_2, \dots, x'_N\}, \quad (2.2)$$

kjer je $\mathbf{x}(t) = \{x_1, x_2, \dots, x_N\}$ vektor koncentracij opazovanih kemijskih zvrsti v časovni točki t in N število opazovanih kemijskih zvrsti. Pri modeliranju dinamike opazujemo

spremembe koncentracij posameznih kemijskih zvrsti, ki se zaradi procesov v sistemu zgodijo v časovnem intervalu dt . Koncentracija opazovane kemijske zvrsti določa njeno prisotnost v biološkem sistemu. Glede na vrsto modela koncentracijo merimo v molih na liter $M = \frac{\text{mol}}{L}$ (deterministični modeli), za katere običajno uporabljamo velikostne razrede μM , nM ali pM , ali pa s številom molekul v sistemu (stohastični modeli). Pri obeh tipih modelov vektor $\mathbf{x}(t) = \{x_1, x_2, \dots, x_N\}$ predstavlja *stanje sistema* v časovni točki t . Zaradi transkripcije in translacije se koncentracije mRNA in proteinov povečujejo, medtem ko se z degradacijo njihove koncentracije zmanjšujejo. Intenzivnost degradacije, transkripcije, pri kateri je potrebno upoštevati vplive transkripcijskih faktorjev, in translacije je spremenljiva, saj so vsi procesi odvisni od trenutnih koncentracij kemijskih zvrsti, ki v reakciji procesa nastopajo, in hitrosti reakcij (angl. *kinetic rate*), ki jih v modelih predstavimo s kinetičnimi parametri ali samo parametri. Slednji so torej poleg trenutnega stanja sistema ključni za kvantitativno simuliranje dinamike bioloških sistemov. Dinamiko biološkega sistema natančno opišemo z matematičnimi modeli.



*Modeliranje dinamike
enostavnih bioloških sistemov*

Modeliranje je pri načrtovanju novih in raziskovanju obstoječih bioloških sistemov ključnega pomena. Pristopi modeliranja dinamike enostavnih bioloških sistemov ali odkrivanja topologije slednjih se razlikujejo glede na njihovo natančnost in računsko zahtevnost ter velikost modela. Izbira pristopa modeliranja je torej kompromis med dostopnostjo podatkov, ki jih o sistemu imamo, in natančnostjo, s katero želimo dinamiko ali topologijo sistema opisati. V pričujočem poglavju predstavimo pregled najpogostejših pristopov k modeliranju bioloških sistemov, ki jih v grobem razdelimo na kvalitativne in kvantitativne [27, 28].

3.1 Kvalitativno modeliranje

S kvalitativnimi modeli opisujemo osnovne lastnosti bioloških sistemov. Razdelimo jih lahko v *topološke pristope*, ki se osredotočajo na odkrivanje novih posrednih in neposrednih interakcij v gensko regulatornih omrežjih, in *dinamične logične pristope*, ki poleg topologije upoštevajo tudi potencialne spremembe v stanju sistema, ki so pri kvalitativnih modelih predstavljene diskretno.

Pri topoloških pristopih se za predstavitev interakcij uporabljajo usmerjeni grafi [29], pri katerih vozlišča predstavljajo posamezne kemijske zvrsti, povezave pa interakcije med njimi. Podajajo nam najbolj osnovni opis sistema, ki pa pogosto služi kot izhodišče za izgradnjo ostalih tako kvalitativnih kot tudi kvantitativnih modelov. Iz topoloških modelov namreč pridobimo podatke o obstoju (prisoten ali neprisoten) in tipu (aktivacija ali represija) vpliva transkripcijskih faktorjev na izražanje posameznih genov v sistemu.

Na podlagi poznavanja interakcij lahko zgradimo dinamični logični sistem, ki poleg topologije opisuje tudi osnovne dinamične lastnosti sistema. *Stanje* v dinamičnem logičnem sistemu določa vektor, ki predstavlja stopnjo izražanja vseh genov v sistemu. Spremembe slednjega opazujemo v diskretnih časovnih korakih, ki nimajo konkretnih časovnih enot. Na podlagi trenutnega stanja sistema in topologije gensko regulatornega omrežja določimo, kako se izražanje posameznih genov v naslednjem koraku spremeni. Najpogosteje se za modeliranje dinamičnih logičnih sistemov uporabljajo Boolove mreže (angl. *Boolean networks*) [30], pri katerih izražanje gena predstavimo z vrednostjo 0 (izražanje je ustavljeno) ali 1 (izražanje je aktivno). Z opazovanjem sprememb v izražanju genov v omrežju se Boolove mreže najpogosteje uporabljajo za odkrivanje potencialno pogosto obiskanih ali celo končnih stanj, v katerih se izražanje genov ne spreminja. Zaradi predstavitve izražanja genov, ki je omejeno na dve vrednosti, Boo-

love mreže niso primerne za preiskovanje bolj natančnih dinamičnih lastnosti omrežja. Za modeliranje slednjih so bolj primerna večvrednostna omrežja (angl. *multi-valued networks*), ki predstavljajo razširitev Boolovih mrež [31]. Izražanje gena je v večvrednostnih omrežjih predstavljeno z več možnimi vrednostmi (npr. ustavljeno, polovično, maksimalno), pri čemer večji nabor vrednosti omogoča bolj natančno analizo dinamičnih lastnosti, a občutno poveča prostor vseh možnih stanj sistema, zato se uporablja le za modeliranje manjših omrežij. Klasične Boolove mreže se modelirajo deterministično, pri čemer na vsakem koraku opazujemo vsa možna stanja, ki jih sistem na tem koraku lahko zavzame. Za stohastično razširitev iskanja stanja, v katerem se sistem najpogosteje nahaja po določenem številu korakov, se uporabljajo verjetnostne Boolove mreže (angl. *probabilistic boolean networks*). Pri slednjih vsaki povezavi v mreži dodelimo neko verjetnost in v vsakem koraku izberemo naključno število genov pri katerih se izražanje spremeni. Spremembe stanj so zato bolj usmerjene k doseganju najbolj verjetnih končnih stanj sistema [32, 33]. Navadne, večvrednostne in verjetnostne Boolove mreže pogosto grafično modeliramo s kvalitativnimi Petrijevim mrežami [34].

Ker je opis dinamičnih lastnosti v kvalitativnih modelih preprost in posledično računsko nezahteven v primerjavi z bolj natančnimi opisi dinamike, se ti pogosto uporabljajo za odkrivanje topologije in intenzivnosti interakcij iz eksperimentalnih podatkov [35–38].

Stanje sistema je pri kvalitativnih modelih predstavljeno s prisotnostjo ali odsotnostjo opazovanih kemijskih zvrsti oziroma z aktivnostjo ali neaktivnostjo opazovanih genov. Za natančnejše modeliranje dinamike sistema uporabljamo kvantitativne pristope. Pri slednjih stanje sistema predstavlja vektor koncentracij opazovanih kemijskih zvrsti. Spremembe koncentracij opazujemo v odvisnosti od časa, ki ga merimo v konkretnih časovnih merskih enotah.

3.2 Kvantitativno modeliranje

Stanje sistema modelov na podlagi kvantitativnih pristopov je predstavljeno z vektorjem koncentracij ali števil molekul vseh opazovanih kemijskih zvrsti $\mathbf{x}(t) = \{x_1(t), x_2(t), \dots, x_N(t)\}$, kjer je N število opazovanih kemijskih zvrsti. Kvantitativne pristope za modeliranje bioloških sistemov razdelimo na *deterministične* in *stohastične*.

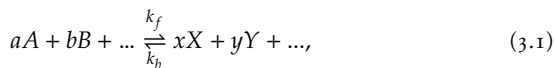
Deterministični pristopi v biološkem sistemu predpostavljajo homogeno okolje, v katerem so kemijske zvrsti po prostoru razporejene enakomerno, v sistemu so prisotne v večjih koncentracijah, vpliv difuzije pa je ničen [39]. S pomočjo njih opazujemo

povprečen odziv biološkega sistema oziroma stanje po daljšem časovnem obdobju, ko difuzija izzveni. V primeru determinističnih pristopov predpostavljamo, da se zaradi velikosti in homogenosti sistema naključnost kemijskih reakcij izniči.

Stohastični pristopi upoštevajo stohastičnost izvajanja kemijskih reakcij, ki je posledica termodinamike in naključnih trkov molekul v sistemu. Uporabljajo se za natančnejše modeliranje dinamike posamezne celice in s tem heterogenost obnašanja populacije identičnih celic, a so zaradi tega računsko bistveno bolj potratni od determinističnih pristopov.

3.2.1 Deterministični pristopi

Izhodišče determinističnih pristopov je zakon o vplivu koncentracij (angl. *law of mass action*) [40], ki temelji na dejstvu, da se masa sistema kot celote s časom ne spreminja in predpostavlja, da ob konstantnih vplivih iz okolice sistem preide v ravnotežno stanje. Stanje sistema je določeno z vektorjem $\mathbf{x}(t) = \{x_1(t), x_2(t), \dots, x_N(t)\}$, kjer je $x_i(t)$ ($i = 1, \dots, N$) realno število, ki predstavlja koncentracijo i -te opazovane kemijske zvrsti. Predpostavimo naslednjo kemijsko reakcijo



kjer $A, B \dots$ predstavljajo *reaktante*, $X, Y \dots$ *produkte*, $a, b, \dots, x, y \dots$ pa *stehiometrične koeficiente*, ki predstavljajo število molekul posamezne zvrsti v uravnoteženi reakciji. Ko sistem preide v ravnotežno stanje veljata izraza

$$k_f[A]^a[B]^b \dots = k_b[X]^x[Y]^y \dots, \quad (3.2)$$

$$K = \frac{k_f}{k_b} = \frac{[X]^x[Y]^y \dots}{[A]^a[B]^b \dots}, \quad (3.3)$$

kjer črke v oglatih oklepajih predstavljajo koncentracijo reagentov oziroma produktov, k_f hitrostno konstantno *vodilne reakcije* (angl. *forward affinity constant*), k_b hitrostno konstanto *povratne reakcije* (angl. *backward affinity constant*), K pa ravnotežno konstanto (angl. *equilibrium constant*). V določenih primerih nas zanima *disociacijska konstanta* (angl. *dissociation constant*), ki predstavlja moč vezi produktov in je definirana z izrazom

$$K_d = \frac{1}{K}. \quad (3.4)$$

Enačbe hitrosti (angl. *rate equation*) vodilne in povratne reakcije zapišemo z izrazoma

$$r_f = k_f[A]^a[B]^b \dots, \quad (3.5)$$

$$r_b = k_b[X]^x[Y]^y \dots, \quad (3.6)$$

ki predstavljata izhodišče za prevedbo kemijskih reakcij v sistem diferencialnih enačb, s katerimi deterministično modeliramo dinamiko sistema. Za demonstracijo prevedbe kemijske reakcije v diferencialno enačbo vzemimo primer preproste enosmerne kemijske reakcije



za katero dobimo *enačbo hitrosti reakcije*

$$r = k_f[A][B], \quad (3.8)$$

kar lahko kot diferencialno enačbo, ki nam podaja spremembo koncentracije produkta C, zapišemo z izrazom

$$\frac{d[C]}{dt} = k_f[A][B]. \quad (3.9)$$

Na podlagi izpeljave enačbe hitrosti reakcij lahko vzpostavimo sistem diferencialnih enačb za vsako od reakcij v biološkem sistemu, ki natančno opisuje njeno dinamiko. Ker v sistemu običajno nastopa veliko kemijskih reakcij in so enačbe lahko nelinearne, model na osnovi enačb hitrosti reakcij pri večjih sistemih hitro postane zelo kompleksen. V ta namen se uporabljajo določene poenostavitve, ki jih predstavimo v naslednjih razdelkih.

Michaelis-Menten kinetika

Michaelis-Menten kinetika opisuje encimske reakcije, vzpostavitev modela pa temelji na zakonu o vplivu koncentracij [41]. Encimsko reakcijo shematično prikažemo z izrazom



kjer E predstavlja encim, S substrat, ES njen kompleks, P produkt reakcije, k_f in k_b hitrostni konstanti vodenja in povratne reakcije tvorbe kompleksa, k_{cat} pa hitrostno konstanto pretvorbe kompleksa ES v produkt P , pri kateri se encim E ne porabi. Iz encimske reakcije v izrazu (3.10) izpeljemo naslednji sistem diferencialnih enačb

$$\frac{d[S]}{dt} = -k_f[E][S] + k_b[ES], \quad (3.11)$$

$$\frac{d[E]}{dt} = -k_f[E][S] + (k_b + k_{cat})[ES], \quad (3.12)$$

$$\frac{d[P]}{dt} = k_{cat}[ES], \quad (3.13)$$

$$\frac{d[ES]}{dt} = k_f[E][S] - (k_b + k_{cat})[ES]. \quad (3.14)$$

Skupno koncentracijo encima $[E_0]$ lahko po zakonu o ohranitvi in ob predpostavki $[S] \gg [E]$ [39, 42] zapišemo z izrazom

$$[E] + [ES] = [E_0]. \quad (3.15)$$

Zaradi hitrosti reakcije tvorbe kompleksa, ki je bistveno višja od hitrosti pretvorbe kompleksa v produkt, lahko pri opazovanju sprememb koncentracije produkta v danem časovnem koraku predpostavimo konstantno koncentracijo kompleksa [43]. Iz izraza (3.14) izpeljemo izraz

$$\frac{d[ES]}{dt} = k_f[E][S] - (k_b + k_{cat})[ES] = 0, \quad (3.16)$$

$$k_f[E][S] = (k_b + k_{cat})[ES]. \quad (3.17)$$

Ob upoštevanju izraza za skupno koncentracijo encima iz izraza (3.15) dobimo izraz

$$k_f([E_0] - [ES])[S] = (k_b + k_{cat})[ES], \quad (3.18)$$

iz katerega s poenostavitvijo izrazimo $[ES]$ in dobimo izraz za hitrosti tvorbe kompleksa

$$r_{ES} = \frac{[E_0][S]}{K_m + [S]}, \quad (3.19)$$

kjer je $K_m = \frac{k_b + k_{cat}}{k_f}$ Michaelisova konstanta. Enačbo hitrosti in pripadajočo diferencialno enačbo tvorbe produkta zapišemo z izrazom

$$r_P = k_{cat}[ES] = \frac{d[P]}{dt} = \frac{V_{max}[S]}{K_m + [S]}, \quad (3.20)$$

kjer je $V_{max} = k_{cat}[E_0]$ maksimalna hitrost tvorbe produkta.

Hillove enačbe

S Hillovimi enačbami deterministično modeliramo proces transkripcije, katere intenzivnost je odvisna od koncentracije transkripcijskih faktorjev [1]. Vezave transkripcijskih faktorjev ne modeliramo eksplcitno, ampak intenzivnost aktivacije oziroma represije določimo neposredno s koncentracijo transkripcijskega faktorja.

V primeru represije je verjetnost prostega promotorja, ki jo smatramo kot verjetnost, da proces transkripcije poteka, določena z razmerjem $[D] : [D_T]$, kjer $[D]$ predstavlja koncentracijo prostih, $[D_T]$ pa število vseh promotorjev in velja

$$[D_T] = [D] + [D_R], \quad (3.21)$$

kjer $[D_R]$ predstavlja koncentracijo promotorjev, na katere je vezan represor. Vezavo in disociacijo represorja in promotorja zapišemo z reakcijo



kjer je R represor, D promotorosko mesto, D_R pa njun kompleks. Ko sistem preide v ravnotežno stanje, velja

$$K_d = \frac{[R][D]}{[D_R]} = \frac{k_b}{k_f}, \quad (3.23)$$

kar lahko zapišemo tudi z izrazom

$$\frac{[D_R]}{[D]} = \frac{[R]}{K_d}. \quad (3.24)$$

Verjetnost prostega promotorja, izraženo s koncentracijo represorja, torej dobimo z združitvijo izrazov (3.21) in (3.24) v izraz

$$\frac{[D]}{[D_T]} = \frac{[D]}{[D] + [D_R]} = \frac{1}{1 + \frac{[D_R]}{[D]}} = \frac{1}{1 + \frac{[R]}{K_d}}, \quad (3.25)$$

čigar vrednost se zmanjšuje z večanjem koncentracije represorja. Sprememba koncentracije $d[mRNA]$ kot posledica procesa transkripcije je definirana kot zmnožek maksimalne aktivnosti promotorja k_{tsc} ter verjetnosti prostega promotorja

$$\frac{d[mRNA]}{dt} = \frac{k_{tsc}}{1 + \frac{[R]}{K_d}}. \quad (3.26)$$

Na podoben način lahko spremembo koncentracije mRNA izpeljemo tudi v primeru vezave aktivatorja na promotor, pri katerem nas zanima razmerje $[D_A] : [D_T]$, kjer $[D_A]$ predstavlja koncentracijo promotorjev, na katere je vezan aktivator, $[D_T]$ pa koncentracijo vseh promotorjev. Spremembo $d[mRNA]$ v primeru aktivacije zapišemo z izrazom

$$\frac{d[mRNA]}{dt} = \frac{k_{tsc}[A]}{K_d + [A]}, \quad (3.27)$$

ki se povečuje z večanjem koncentracije aktivatorja.

V določenih primerih se na promotorska mesta vežejo le transkripcijski faktorji, ki so povezani v večje molekule (npr. dimere, tetramere itd.). V tem primeru bi bilo potrebno pri modeliranju z enačbami hitrosti reakcij definirati tudi vodilne in povratne reakcije za združitve transkripcijskih faktorjev v večje molekule. Z uporabo Hillovih enačb se temu ognemo, saj slednje predpostavljajo, da reakcija združitve poteka istočasno kot vezava na promotor. Predpostavka Hillove enačbe temelji na principu kooperativne vezave. Za slednjega velja, da pri kooperativnosti večji od 1 (veže se lahko več kot 1 molekula) vezava ene molekule poveča afiniteto vezave ostalih molekul. V tem primeru je verjetnost vezave zgolj ene molekule bistveno manjša od verjetnosti vezave več molekul zato se predpostavlja, da se vse molekule vežejo skupaj. Ker se pri eni vezavi transkripcijskega faktorja na promotorsko mesto porabi več molekul transkripcijskega faktorja, se temu primerno spremeni stehiometrični koeficient pri reakciji vezave na promotor. Slednji se v diferencialnih enačbah za spremembo koncentracije

mRNA prenese v eksponente. Spremembi koncentracije mRNA za primera represije in aktivacije izražanja v tem primeru zapišemo z izrazoma

$$\frac{d[mRNA]}{dt} = \frac{k_{tsc}}{1 + \left(\frac{[R]}{K_d}\right)^n}, \quad (3.28)$$

$$\frac{d[mRNA]}{dt} = \frac{k_{tsc}[A]^n}{K_d^n + [A]^n}, \quad (3.29)$$

kjer je n Hillov koeficient in določa število molekul transkripcijskega faktorja, ki se morajo združiti pred vezavo na promotor.

3.2.2 Stohastični pristopi

Stohastični pristopi v nasprotju z determinističnimi upoštevajo tudi stohastičnost kemijskih reakcij, ki je posledica termodinamičnih lastnosti sistema. Stohastično modeliranje se uporablja pri manjših sistemih, v katerih so koncentracije opazovanih kemijskih zvrsti relativno majhne in so zato posledice naključnih trkov molekul bolj izražene. Ker opazujemo število individualnih molekul, je stanje sistema za razliko od determinističnih pristopov predstavljeno z vektorjem $\mathbf{x}(t) = \{x_1(t), x_2(t), \dots, x_N(t)\}$, pri katerem je $x_i(t)$ ($i = 1, \dots, N$) celo število, ki predstavlja število molekul i -te opazovane kemijske zvrsti.

Stohastični model sestavlja M različnih reakcij r_j ($j = 1, \dots, M$), na podlagi katerih se spreminja stanje sistema. Sprememba stanja je določena z izbiro reakcije, ki je v modelu zapisana v obliki stehiometričnega vektorja s_j dolžine N . Slednji vsebuje stehiometrične koeficiente kemijskih zvrsti, ki nastopajo v reakciji r_j . Ob izbiri reakcije se stehiometrični vektor prišteje vektorju $\mathbf{x}(t)$, s čimer se število molekul kemijskih zvrsti, ki v izbrani reakciji nastopajo kot reagenti, zmanjša, število molekul, ki v izbrani reakciji nastopajo kot produkti, pa poveča. Spremembo stanja ob poteku reakcije r_j je podana z izrazom

$$\mathbf{x}(t + dt) = \mathbf{x}(t) + s_j, \quad (3.30)$$

kjer s_j predstavlja stehiometrični vektor reakcije r_j . Stehiometrične vektorje reakcij združimo v stehiometrično matriko S . Izbira reakcije, ki se bo v danem časovnem koraku izvršila, je odvisna od števila molekul reaktantov in od vrednosti, ki predstavlja

hitrost reakcije. Slednja je sicer osnovana na pripadajoči reakciji determinističnega modela, a se glede na tip reakcije in dani volumen celice lahko spremeni. Verjetnost izbire reakcije r_j izračunamo s funkcijo $a_j(\mathbf{x}(t))$, katere oblika je odvisna od tipa reakcije. Glede na število in tip nastopajočih reaktantov ločimo več vrst funkcij, ki so opisane v virih [44–46].

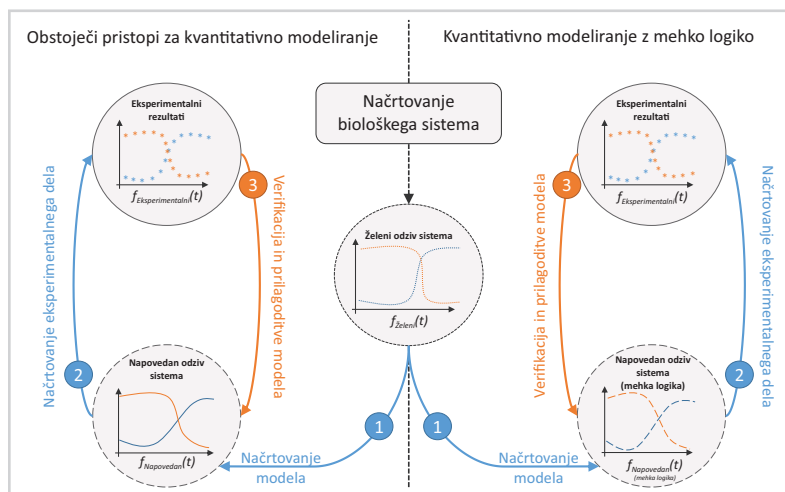
Za pridobitev simulacijskih rezultatov pri stohastičnih modelih poznamo več pristopov. Za analitično določitev vrednosti $a_j(\mathbf{x}(t))$ se pri majhnih in enostavnih sistemih uporablja glavna kemijska enačba (angl. *Chemical Master Equation* - CME). Z večanjem sistema se število reakcij bistveno poveča, zato problem analitičnega reševanja glavne kemijske enačbe hitro postane neobvladljiv. Za numerično reševanje stohastičnih modelov se najpogosteje uporabljata Gillespiejev algoritem za stohastično simuliranje (angl. *Stochastic Simulation Algorithm* - SSA) [47] in metoda τ -skokov (angl. τ -leaping) [48, 49]. Obe metodi za numerično reševanje uporabljata generiranje naključnih števil. Ena simulacija torej izračuna le eno od možnih rešitev. Obe metodi sta računsko bolj potratni od determinističnih, poleg tega pa je potrebno za pridobitev povprečnega odziva biološkega sistema izvesti veliko število simulacij.

3.3 Omejitve obstoječih pristopov

Kvalitativni pristopi se uporabljajo za osnovno razumevanje sistema, topologije njegovega gensko regulatornega omrežja ali dinamičnih lastnosti sistema, niso pa primerne za natančno modeliranje dinamike, saj se pri pridobivanju simulacijskih rezultatov ne uporablja natančnih vrednosti kinetičnih parametrov in rezultati posledično niso predstavljeni s konkretnimi merskimi enotami. Spremembe koncentracij, izražene v primernih enotah v odvisnosti od časa, nam podajo kvantitativni modeli. Slednji sicer dinamiko opišejo bolj natančno, a tako deterministični kot tudi stohastični modeli za pridobitev simulacijskih rezultatov potrebujejo poznavanje vrednosti kinetičnih parametrov, ki opisujejo hitrosti reakcij.

V pričujoči doktorski disertaciji predstavimo metodo, ki za modeliranje uporablja mehko logiko. Slednja dinamiko opiše na lingvističen način in nam omogoča, da kljub neznanim natančnim vrednostim kinetičnih parametrov odziv modela še vedno predstavimo kvantitativno. Spremembe koncentracij opazovanih kemijskih zvrsti tako kot pri obstoječih kvantitativnih metodah izrazimo s konkretnimi merskimi enotami v odvisnosti od časa. Slika 3.1 prikazuje podobnost obstoječih kvantitativnih pristopov in metode, predstavljene v pričujoči disertaciji, ki nam omogoča pridobitev relevantnih

kvantitativnih rezultatov modeliranja in s tem pospešitev procesa načrtovanja novega ali analizo obstoječega biološkega sistema, tudi kadar vrednosti kinetičnih parametrov niso znane.



Slika 3.1

Postopek pridobivanja simulacijskih rezultatov je s predlagano metodo (desni del sheme) ekvivalenten postopku obstoječih metod (levi del sheme). Prednost naše metode je v možnosti izvedbe prvega koraka, tudi kadar natančnih vrednosti kinetičnih parametrov ne poznamo.

Predlagana metoda modeliranja je združljiva z obstoječimi kvantitativnimi determinističnimi metodami. Uporabimo jo lahko kot dopolnitev modela v delih, kjer vrednosti kinetičnih parametrov niso znane, ter s tem ohranimo natančnost obstoječih determinističnih metod v preostalih delih.



Osnove mehke logike

V pričujočem poglavju predstavimo osnovne koncepte *mehke logike*, ki jih kasneje uporabimo pri gradnji modelov dinamike bioloških sistemov. Definiramo pojma *mehke množice* in *mehke spremenljivke*. Na preprostem primeru uporabe mehke logike za potrebe krmiljenja ambientalne temperature predstavimo odločanje na osnovi *vrednosti mehkih spremenljivk* ter tako opišemo mehčanje, uporabo IF-THEN pravil (inferenca) in ostrenje ter na kratko primerjamo uporabo mehkih množic in mehke logike s klasično teorijo množic. Način zapisa in terminologijo smo povzeli po enem od pionirjev mehke logike H.-J. Zimmermannu [50].

4.1 Mehka množica

V *klasični logiki* element x prostora U pripada opazovani množici A ($x \in A$), ali pa tej množici ne pripada ($x \notin A$). Takšna obravnava pripadnosti je velikokrat zadovoljiva, (npr. v primeru množice negativnih števil A , za katero velja ($x \in A \mid x < 0$) oziroma ($x \notin A \mid x \geq 0$)), a v določenih primerih neustrezna. Za primer vzemimo množico visokih ljudi A , katere pripadniki so osebe, visoke 180 cm ali več, in tri osebe, ki so po vrsti visoke 179 cm (oseba x_1), 181 cm (oseba x_2) in 199 cm (oseba x_3). Čeprav sta si po višini osebi x_1 in x_2 bistveno bolj podobni kot osebi x_2 in x_3 , slednji množici A po njeni ostri definiciji pripadata, oseba x_1 pa ne.

V *mehki logiki*, ki predstavlja razširitev klasične logike, je pripadnost elementa množici lahko tudi delna. Množico, ki ji elementi lahko pripadajo tudi delno, imenujemo *mehka množica*. V nadaljevanju po [50] povzemamo definicijo mehke množice.

Definicija 1: Mehka množica \tilde{A} elementov x iz prostora U je množica urejenih parov

$$\tilde{A} = \{(x, \mu_{\tilde{A}}(x)) \mid x \in U\}, \quad (4.1)$$

kjer je $\mu_{\tilde{A}}$ *pripadnostna funkcija* mehke množice \tilde{A} , na podlagi katere izračunamo *stopnjo pripadnosti* posameznega elementa x k mehki množici \tilde{A} ($\mu_{\tilde{A}}(x) \in [0, 1]$).

Stopnja pripadnosti elementa x mehki množici \tilde{A} je torej vrednost, ki jo dobimo s preslikavo na osnovi pripadnostne funkcije, pri čemer vrednosti x iz prostora U ($x \in U$) preslikamo v zvezni interval $[0, 1]$ po izrazu

$$\mu_{\bar{A}} : U \rightarrow [0, 1], \quad (4.2)$$

medtem ko je pripadnost klasični množici A preslikava vrednosti x iz prostora U ($x \in U$) v nabor vrednosti $\{0, 1\}$ po izrazu

$$\mu_A : U \rightarrow \{0, 1\}. \quad (4.3)$$

4.2 Mehka spremenljivka

Numeričnim spremenljivkam običajno prirejamo natančne numerične vrednosti, ki jih imenujemo tudi *ostre* (angl. *crisp*) vrednosti. Spremenljivko, katere vrednosti ne predstavimo ostro, temveč s pripadnostmi k mehkim množicam, imenujemo *mehka spremenljivka*. V nadaljevanju po [50] povzemamo definicijo mehke spremenljivke.

Definicija 2: Mehka spremenljivka je določena s trojico (S, U, T) . Pomeni posameznih entitet trojice so sledeči:

- S predstavlja ime spremenljivke (npr. *Age*),
- U predstavlja zalogo ostrih vrednosti, nad katerimi je opazovana mehka spremenljivka S definirana (npr. interval $U = [0, 100]$),
- T predstavlja nabor mehkih množic $\tilde{A}_1, \tilde{A}_2, \dots, \tilde{A}_n$, ki so definirane nad U , s pomočjo katerih predstavimo *vrednosti mehke spremenljivke* S .

Vrednosti mehke spremenljivke S predstavimo s pripadnostmi k posameznim mehkim množicam. Slednje lahko poimenujemo indeksirano $(\tilde{A}_1, \tilde{A}_2, \dots, \tilde{A}_n)$, ali pa z lingvističnimi izrazi (npr. $\widetilde{\text{YOUNG}}, \widetilde{\text{MATURE}}, \widetilde{\text{OLD}}$), zaradi česar mehko spremenljivko imenujemo tudi *lingvistična spremenljivka*.

Za primer vzpostavimo mehko spremenljivko *Age*, ki je definirana na intervalu od 0 do 100 let ($U = [0, 100]$). Predpostavimo, da ta interval pokriva nabor mehkih množic $T = \{\widetilde{\text{YOUNG}}, \widetilde{\text{MATURE}}, \widetilde{\text{OLD}}\}$. Pripadnostne funkcije mehkih množic, za katere velja, da je njihov maksimum po vrednosti 1 ($\forall x \in U, i : 1 \leq i \leq n \mid 0 \leq \mu_{\tilde{A}_i}(x) \leq 1$), so lahko poljubne oblike, a najpogosteje uporabljamo trikotne in trapezoidne funkcije.

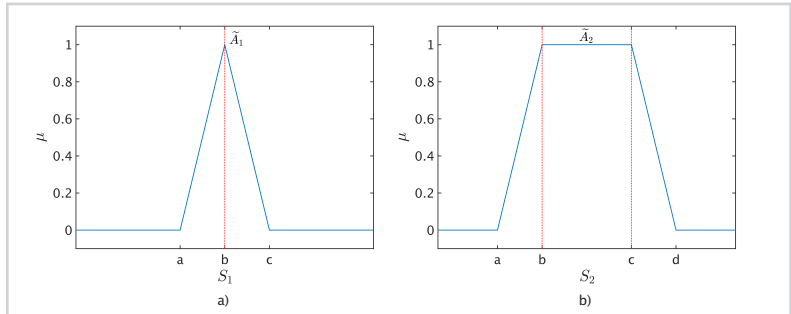
Trikotno funkcijo podamo s parametričnim zapisom $[a, b, c]_p$, trapezoidno pa s parametričnim zapisom $[a, b, c, d]_p$ ($\{a, b, c, d\} \in U$). Pripadnost $\mu_{\tilde{A}}$ mehki množici \tilde{A} se pri trikotni pripadnostni funkciji, podani s parametričnim zapisom $[a, b, c]_p$ izračuna na podlagi izraza

$$\mu_{\tilde{A}}(x) = \begin{cases} 0, & x \leq a, \\ \frac{x-a}{b-a}, & a \leq x \leq b, \\ \frac{c-x}{c-b}, & b \leq x \leq c, \\ 0, & x > c, \end{cases} \quad (4.4)$$

pri trapezoidni pripadnostni funkciji, podani s parametričnim zapisom $[a, b, c, d]_p$ pa na podlagi izraza

$$\mu_{\tilde{A}}(x) = \begin{cases} 0, & x \leq a \\ \frac{x-a}{b-a}, & a \leq x \leq b, \\ 1, & b \leq x \leq c, \\ \frac{d-x}{d-c}, & c \leq x \leq d, \\ 0, & x > d. \end{cases} \quad (4.5)$$

Parametrični zapis pripadnostnih funkcij smo povzeli po programskem orodju MATLAB Simulink® [51]. Primera omenjenih pripadnostnih funkcij sta prikazana na sliki 4.1.



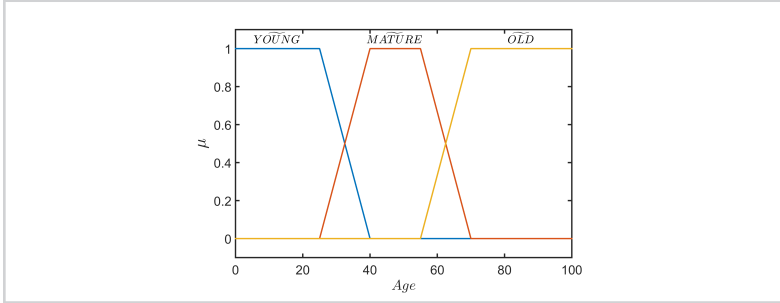
Slika 4.1

Primer definicije trikotne (a) in trapezoidne (b) pripadnostne funkcije mehkih množic \tilde{A}_1 in \tilde{A}_2 , ki opisujeta mehki spremenljivki S_1 in S_2 .

Predpostavimo, da za spremenljivko *Age* določimo trapezoidne pripadnostne funkcije po izrazih

$$\mu_{\widetilde{\text{YOUNG}}} = [0, 0, 25, 40]_p, \mu_{\widetilde{\text{MATURE}}} = [25, 40, 55, 70]_p, \mu_{\widetilde{\text{OLD}}} = [55, 70, 100, 100]_p, \quad (4.6)$$

katerih grafično predstavitev predstavlja slika 4.2.



Slika 4.2

Mehka spremenljivka *Age* s pripadajočimi trapezoidnimi pripadnostnimi funkcijami mehkih množic YOUNG, MATURE in OLD.

Vrednost opazovane ostre spremenljivke iz intervala U lahko mehko opišemo s pripadnostmi k posameznim mehkim množicam iz T . V primeru ostre vrednosti spremenljivke *Age* pri 28 letih ($x = 28$) bi tako dobili pripadnosti po izrazu

$$\mu_{\widetilde{\text{YOUNG}}}(28) = 0.8, \mu_{\widetilde{\text{MATURE}}}(28) = 0.2, \mu_{\widetilde{\text{OLD}}}(28) = 0. \quad (4.7)$$

4.3 Mehki odločitveni sistem

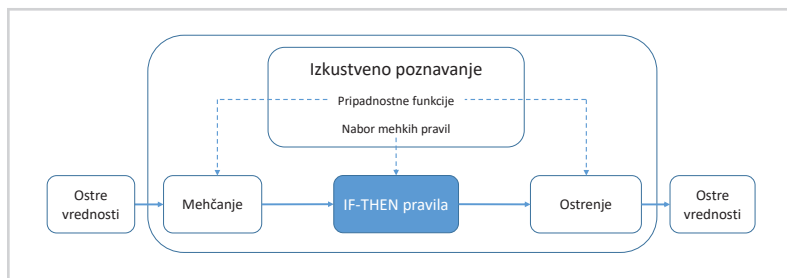
Sistem, ki za odločanje (sklepanje, decizijo ali inferenco) uporablja vrednosti mehkih spremenljivk, imenujemo *mehki odločitveni sistem*. Postopek odločanja, ki vodi do posameznih odločitev skozi čas, se vrši na podlagi izkustvenega znanja, ki ga v sistem vgradimo v obliki množice lingvističnih IF-THEN pravil. Ta določajo, kako se glede na vrednosti vhodnih mehkih spremenljivk oblikujejo vrednosti ene ali večih izhodnih mehkih spremenljivk. V mehki odločitveni sistem običajno vstopajo in izstopajo ostre vrednosti spremenljivk, za potrebe sklepanja pa v notranjosti sistema nastopajo v mehki obliki.

Za sklepanje v mehkem odločitvenem sistemu uporabljamo vrednosti mehkih spremenljivk, zato ostre vrednosti spremenljivk na vходу v sistem *mehčamo* (angl. *fuzzification*). Na podlagi vrednosti vhodnih mehkih spremenljivk nato s postopkom odločanja pridobimo vrednosti izhodnih mehkih spremenljivk, ki predstavljajo posamezno

odločitev ali sklep mehkega sistema. Želeni rezultat mehkega odločitvenega sistema je ponavadi ena ali več ostrih vrednosti na izhodu, zato pred izstopom iz sistema vrednosti izhodnih mehkih spremenljivk *ostrimo* (angl. *defuzzification*). Oris delovanja mehkega odločitvenega sistema prikazuje slika 4.3.

Slika 4.3

Prikaz delovanja mehkega odločitvenega sistema. Vanj vstopajo ostre vrednosti vhodnih spremenljivk, ki se po pravilih mehčanja pretvorijo v vrednosti mehkih spremenljivk. Te se nato uporabijo pri odločanju na podlagi definiranih IF-THEN pravil. Z odločanjem pridobimo vrednosti izhodnih mehkih spremenljivk, ki iz sistema po ostrenju izstopijo kot ostre vrednosti.



Kot primer mehkega odločitvenega sistema v naslednjem razdelku predstavimo mehki krmilnik za nadzor ambientalne temperature.

4.4 Vzorčni primer mehkega odločitvenega sistema

Predpostavimo, da se nahajamo pred problemom izdelave logike krmiljenja ogrevalnega sistema, ki naj bi zagotavljal želeno ambientalno temperaturo prostora. Ob predpostavki, da ne poznamo natančnega matematičnega modela, ki bi opisoval spremembo ambientalne temperature v odvisnosti od spremembe intenzivnosti ogrevanja, krmilnik realiziramo kot mehki odločitveni sistem. Krmilnik v diskretnih korakih zajema trenutno ambientalno temperaturo, iz katere lahko izračunamo tudi spremembo temperature med dvema zaporednima meritvama. Na podlagi zajetih in izračunanih podatkov ustrezno reguliramo moč ogrevanja, s čimer dosežemo želeno ambientalno temperaturo prostora. Za opis delovanja mehkega krmilnika določimo tri ostre spremenljivke, in sicer

- vhodno ostro spremenljivko $T_a(t)$, ki predstavlja vrednost ambientalne temperature v časovni točki t ,
- vhodno ostro spremenljivko $\Delta T_a(t)$, ki predstavlja spremembo ambientalne temperature v časovni točki t ($\Delta T_a(t) = T_a(t) - T_a(t-1)$) in

- izhodno ostro spremenljivko $Out(t)$, ki predstavlja intenzivnost ogrevanja v časovni točki t .

Za potrebe mehkega krmilnika moramo definirati vhodne mehke spremenljivke (število mehkih množic in oblike njihovih pripadnostnih funkcij), IF-THEN pravila, na podlagi katerih se vrši odločanje, izhodne mehke spremenljivke ter način ostrenja izhodne spremenljivke Out . Mehčanje, množico IF-THEN pravil in ostrenje povzemamo po primeru navedenemu v viru [50].

4.4.1 Določanje mehkih spremenljivk

Pri določitvi načina mehčanja ostrih vrednosti moramo za obe vhodni ostri spremenljivki določiti njuno mehko obliko s trojico (S, U, T) . Temperatura $T_a(t)$ je na vходу krmilnika izražena numerično v °C. Predpostavimo, da bo njena ostra vrednost v intervalu $U = [0\text{ °C}, 40\text{ °C}]$. Izbira primerne števila mehkih množic (velikost nabora T), ki ta interval pokrivajo, je odvisna od želene kvalitete krmiljenja. Večje število mehkih množic kvaliteto krmiljenja izboljša, a hkrati poveča tudi število IF-THEN pravil, ki jih je potrebno ovrednotiti, kar posledično povečuje procesno zahtevnost odločanja [52]. Predpostavimo, da za vhodno mehko spremenljivko, ki predstavlja trenutno ambientalno temperaturo in jo poimenujemo $Temp$, zadostuje nabor mehkih množic $T = \{\text{VERY LOW}, \text{LOW}, \text{COMFORTABLE}, \text{HIGH}, \text{VERY HIGH}\}$. Za drugo vhodno mehko spremenljivko, ki predstavlja spremembo temperature in jo poimenujemo $Delta_temp$, predpostavimo, da bo njena ostra vrednost v intervalu $U = [-2\text{ °C}, 2\text{ °C}]$ in da za njen opis zadostuje nabor mehkih množic $T = \{\text{NEGATIVE BIG}, \text{NEGATIVE SMALL}, \text{ZERO}, \text{POSITIVE SMALL}, \text{POSITIVE BIG}\}$. Oblike pripadnostnih funkcij posameznih mehkih množic običajno določimo na podlagi izkustvenega znanja o problemski domeni. Predpostavimo, da za vhodno mehko spremenljivko $Temp$ določimo pripadnostne funkcije posameznih mehkih množic po izrazu

$$\begin{aligned}\mu_{\widetilde{VL}} &= [0, 0, 9, 14]_p, \mu_{\widetilde{L}} = [10, 15, 20]_p, \mu_{\widetilde{C}} = [16, 21, 26]_p, \\ \mu_{\widetilde{H}} &= [22, 27, 32]_p, \mu_{\widetilde{VH}} = [28, 33, 40, 40]_p,\end{aligned}\tag{4.8}$$

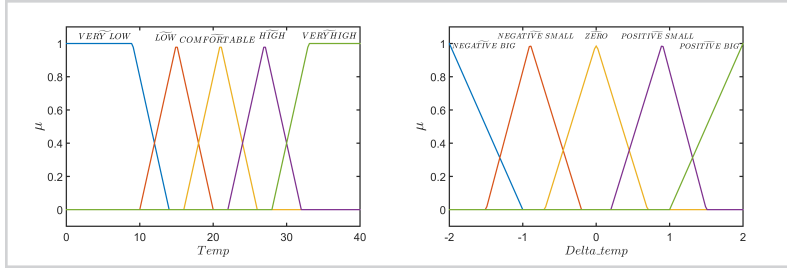
za vhodno mehko spremenljivko $Delta_temp$ pa po izrazu

$$\begin{aligned}\mu_{\widetilde{NB}} &= [-2, -2, -1]_p, \mu_{\widetilde{NS}} = [-1.5, -0.9, -0.2]_p, \mu_{\widetilde{Z}} = [-0.7, 0, 0.7]_p, \\ \mu_{\widetilde{PS}} &= [0.2, 0.9, 1.5]_p, \mu_{\widetilde{PB}} = [1, 2, 2]_p.\end{aligned}\quad (4.9)$$

Mehke množice in njihove pripadnostne funkcije za vhodni mehki spremenljivki *Temp* in *Delta_temp* so prikazane na sliki 4.4.

Slika 4.4

Definicija vhodnih mehkih spremenljivk *Temp* in *Delta_temp*. Iz grafov so razvidni interval U (x os), nabor mehkih množic T (imena nad pripadnostnimi funkcijami) in oblika pripadnostnih funkcij posameznih mehkih množic.

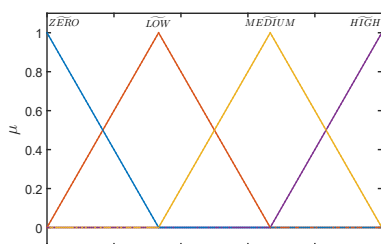


Intenzivnost delovanja ogrevalnega sistema določa izhodna ostra spremenljivka $Out(t)$. Predpostavljamo, da krmilnik omogoča zvezno nastavljanje intenzivnosti ogrevanja, zato za zalogo ostre vrednosti spremenljivke $Out(t)$ na izhodu določimo zvezni interval $U = [0 \%, 100 \ %]$. Vrednost ostre izhodne spremenljivke bomo dobili z ostrenjem vrednosti izhodne mehke spremenljivke, ki jo pridobimo s procesom odločanja. Predpostavimo, da za potrebe krmiljenja ogrevalnega sistema za izhodno mehko spremenljivko, ki jo poimenujemo *Heat_power*, zadostuje nabor mehkih množic $T = \{\widetilde{ZERO}, \widetilde{LOW}, \widetilde{MEDIUM}, \widetilde{HIGH}\}$, za katere določimo pripadnostne funkcije po izrazu

$$\begin{aligned}\mu_{\widetilde{Z}} &= [0, 0, 35]_p, \mu_{\widetilde{L}} = [0, 35, 65]_p, \mu_{\widetilde{M}} = [35, 65, 100]_p, \\ \mu_{\widetilde{H}} &= [65, 100, 100]_p.\end{aligned}\quad (4.10)$$

Mehke množice in njihove pripadnostne funkcije za izhodno mehko spremenljivko *Heat_power* so prikazane na sliki 4.5.

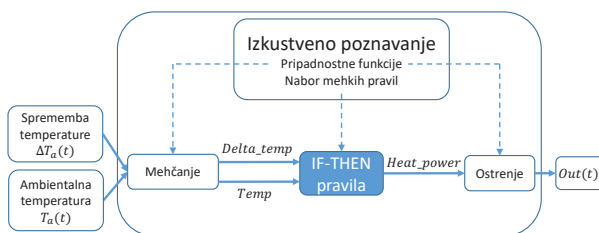
Shemo krmilnika za nadzor ambientalne temperature prikazuje slika 4.6.



Slika 4.5

Definicija izhodne mehke spremenljivke *Heat_power*, ki vključuje definicijo intervala U , nabora T in oblike

po-
o-
žic.



Slika 4.6

Prikaz sheme mehkega krmilnika za krmiljenje ogrevalnega sistema. Vanj vstopata ostri spremenljivki $T_a(t)$ (trenutna ambientalna temperatura) in $\Delta T_a(t)$ (sprememba ambientalne temperature), ki jih z mehčanjem pretvorimo v mehki spremenljivki *Temp* in *Delta_temp*. Na podlagi IF-THEN pravil dobimo mehko vrednost izhodne mehke spremenljivke *Heat_power*, ki jo z ostrenjem pretvorimo v ostro vrednost izhodne ostre spremenljivke *Out(t)*, na podlagi katere se nastavi intenzivnost ogrevanja.

4.4.2 Mehko odločanje

Odločanje v mehkem odločitvenem sistemu imenujemo *mehko odločanje*. Slednje temelji na *mehkih IF-THEN pravilih*. V pogojnem delu mehkega pravila nastopajo vhodne mehke spremenljivke, v sklepnem delu pa izhodne mehke spremenljivke. Kadar v pogojnem delu nastopa več spremenljivk, jih lahko povežemo s poljubno logično operacijo. Najpogosteje uporabljamo operaciji AND ali OR. Mehko pravilo r , v katerem so vhodne mehke spremenljivke S_i ($i = 1, \dots, n$) povezane z operacijo AND, zapišemo z izrazom

$$r: \text{IF } (S_1 \text{ is } \widetilde{A}_j^1) \text{ AND } (S_2 \text{ is } \widetilde{A}_j^2) \text{ AND...AND } (S_n \text{ is } \widetilde{A}_j^n) \text{ THEN } (Y \text{ is } \widetilde{B}_j), \quad (4.11)$$

kjer je \widetilde{A}_j^i j -ta mehka množica vhodne mehke spremenljivke S_i in \widetilde{B}_j mehka množica izhodne mehke spremenljivke Y . Na podlagi vrednosti vhodnih mehkih spremenljivk za vsako pravilo izračunamo *stopnjo izpolnjenosti pogojnega dela pravila* [53]. Stopnjo izpolnjenosti pravila r , pri katerem spremenljivke S_i povežemo z operacijo AND, dobimo po izrazu

$$\alpha_r = \min_{i=1, \dots, n} \{\mu_j^i(S_i^{input})\}. \quad (4.12)$$

V primeru, da bi spremenljivke S_i povezali z operacijo OR, stopnjo izpolnjenosti pravila r dobimo po izrazu

$$\alpha_r = \max_{i=1, \dots, n} \{\mu_j^i(S_i^{input})\}. \quad (4.13)$$

Maksimalno število mehkih pravil je določeno s številom vhodnih mehkih spremenljivk in z velikostmi naborov mehkih množic T za posamezno vhodno mehko spremenljivko. Stopnjo izpolnjenosti izračunamo po izrazih (4.12) oziroma (4.13) za vsa pravila. *Sklepe* posameznega pravila izračunamo po izrazu

$$\mu_r^{\widetilde{B}}(u) = \min\{\alpha_r, \mu_{\widetilde{B}}(u)\}. \quad (4.14)$$

Končno vrednost izhodne mehke spremenljivke, ki je predstavljena s pripadnostmi k posameznim mehkim množicam izhodne mehke spremenljivke, dobimo z združitvijo sklepov $\mu_r^{\widetilde{B}}(u)$ vseh pravil po izrazu

$$\mu(u) = \max_r \{\mu_r^{\widetilde{B}}(u)\}. \quad (4.15)$$

Pri mehkem krmilniku ogrevalnega sistema za vzdrževanje ambientalne temperature na podlagi dveh vhodnih mehkih spremenljivk *Temp* in *Delta_temp* odločamo kakšna naj bo intenzivnost ogrevanja, ki je v krmilniku definirana kot izhodna mehka spremenljivka *Heat_power*. Primer enega izmed pravil, ki v krmilniku vplivajo na

odločanje, je pravilo, pri katerem se ob majhnem padanju temperature in udobni trenutni ambientalni temperaturi intenzivnost ogrevanja nastavi na srednjo vrednost. To pravilo kot IF-THEN pravilo zapišemo z izrazom

$$r_{12}: \text{ IF } (Temp \text{ is } \widetilde{\text{COMFORTABLE}}) \text{ AND } (Delta_temp \text{ is } \widetilde{\text{NEGATIVE SMALL}}) \\ \text{ THEN } (Heat_power \text{ is } \widetilde{\text{MEDIUM}}). \quad (4.16)$$

Kadar se na podlagi vrednosti dveh vhodnih mehkih spremenljivk odločamo, kakšna bo vrednost izhodne mehke spremenljivke, lahko pravila strnjeno zapišemo v tabelo pravil. Tabela 4.1 prikazuje pravila, ki smo jih definirali za mehki krmilnik ogrevalnega sistema.

Tabela 4.1

Strnjen zapis IF-THEN pravil, na podlagi katerih sistem tvori odločitev o vrednosti izhodne mehke spremenljivke *Heat_power*.

<i>Temp/Delta_temp</i>	$\widetilde{\text{NB}}$	$\widetilde{\text{NS}}$	$\widetilde{\text{Z}}$	$\widetilde{\text{PS}}$	$\widetilde{\text{PB}}$
$\widetilde{\text{VL}}$	$\widetilde{\text{HIGH}}$	$\widetilde{\text{HIGH}}$	$\widetilde{\text{HIGH}}$	$\widetilde{\text{MEDIUM}}$	$\widetilde{\text{MEDIUM}}$
$\widetilde{\text{L}}$	$\widetilde{\text{HIGH}}$	$\widetilde{\text{MEDIUM}}$	$\widetilde{\text{MEDIUM}}$	$\widetilde{\text{NIZKA}}$	$\widetilde{\text{NIZKA}}$
$\widetilde{\text{C}}$	$\widetilde{\text{MEDIUM}}$	$\widetilde{\text{MEDIUM}}$	$\widetilde{\text{LOW}}$	$\widetilde{\text{LOW}}$	$\widetilde{\text{ZERO}}$
$\widetilde{\text{H}}$	$\widetilde{\text{MEDIUM}}$	$\widetilde{\text{LOW}}$	$\widetilde{\text{LOW}}$	$\widetilde{\text{ZERO}}$	$\widetilde{\text{ZERO}}$
$\widetilde{\text{VH}}$	$\widetilde{\text{MEDIUM}}$	$\widetilde{\text{LOW}}$	$\widetilde{\text{LOW}}$	$\widetilde{\text{ZERO}}$	$\widetilde{\text{ZERO}}$

Za ponazoritev delovanja krmilnika ogrevalnega sistema predpostavimo, da sta vrednosti vhodnih ostrih spremenljivk $T_a(t) = 23^\circ\text{C}$ in $\Delta T_a(t) = -0.6^\circ\text{C}$. Glede na pravila mehčanja, ki smo jih opisali v razdelku 4.4.1, je vrednost vhodne mehke spremenljivke *Temp*

$$\mu_{\widetilde{\text{C}}}(23^\circ\text{C}) = 0.6, \mu_{\widetilde{\text{H}}}(23^\circ\text{C}) = 0.2, \quad (4.17)$$

vrednost vhodne mehke spremenljivke *Delta_temp* pa

$$\mu_{\widetilde{\text{NS}}}(-0.6^\circ\text{C}) = 0.57, \mu_{\widetilde{\text{Z}}}(-0.6^\circ\text{C}) = 0.14. \quad (4.18)$$

Mehke množice $\widetilde{\text{COMFORTABLE}}$ in $\widetilde{\text{HIGH}}$ spremenljivke *Temp* ter $\widetilde{\text{NEGATIVE SMALL}}$ in $\widetilde{\text{ZERO}}$ spremenljivke *Delta_temp* se pojavijo v naslednjih pravilih

$$\begin{aligned}
r_{12}: & \text{ IF } (Temp \text{ is } \widetilde{\text{COMFORTABLE}}) \text{ AND } (Delta_temp \text{ is } \widetilde{\text{NEGATIVE SMALL}}) \\
& \text{ THEN } (Heat_power \text{ is } \widetilde{\text{MEDIUM}}), \\
r_{13}: & \text{ IF } (Temp \text{ is } \widetilde{\text{COMFORTABLE}}) \text{ AND } (Delta_temp \text{ is } \widetilde{\text{ZERO}}) \\
& \text{ THEN } (Heat_power \text{ is } \widetilde{\text{LOW}}), \\
r_{17}: & \text{ IF } (Temp \text{ is } \widetilde{\text{HIGH}}) \text{ AND } (Delta_temp \text{ is } \widetilde{\text{NEGATIVE SMALL}}) \\
& \text{ THEN } (Heat_power \text{ is } \widetilde{\text{LOW}}), \\
r_{18}: & \text{ IF } (Temp \text{ is } \widetilde{\text{HIGH}}) \text{ AND } (Delta_temp \text{ is } \widetilde{\text{ZERO}}) \\
& \text{ THEN } (Heat_power \text{ is } \widetilde{\text{LOW}}).
\end{aligned}$$

Na podlagi pravil dobimo sledeče stopnje izpolnjenosti za posamezno pravilo

$$\begin{aligned}
\alpha_{12} &= \min\{0.6, 0.57\} = 0.57, \\
\alpha_{13} &= \min\{0.6, 0.14\} = 0.14, \\
\alpha_{17} &= \min\{0.2, 0.57\} = 0.2, \\
\alpha_{18} &= \min\{0.2, 0.14\} = 0.14.
\end{aligned} \tag{4.19}$$

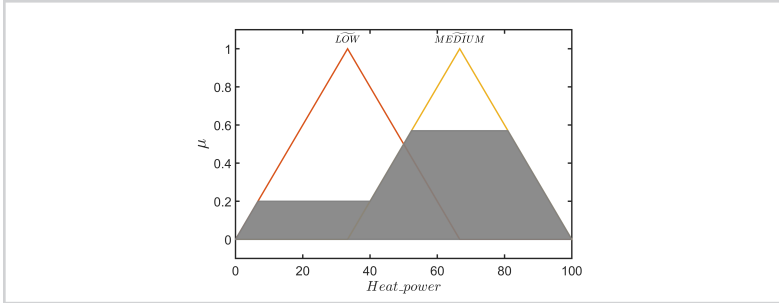
Sklepi posameznih pravil so

$$\begin{aligned}
\mu_{12}^{\widetilde{\text{MEDIUM}}}(u) &= \min\{0.57, \mu_{\widetilde{\text{MEDIUM}}}(u)\}, \\
\mu_{13}^{\widetilde{\text{LOW}}}(u) &= \min\{0.14, \mu_{\widetilde{\text{LOW}}}(u)\}, \\
\mu_{17}^{\widetilde{\text{LOW}}}(u) &= \min\{0.2, \mu_{\widetilde{\text{LOW}}}(u)\}, \\
\mu_{18}^{\widetilde{\text{LOW}}}(u) &= \min\{0.14, \mu_{\widetilde{\text{LOW}}}(u)\}.
\end{aligned} \tag{4.20}$$

Končno vrednost izhodne mehke spremenljivke *Heat_power* dobimo z združitvijo sklepov. Ker se sklepi pravil r_{13} , r_{17} in r_{18} nanašajo na isto mehko množico ($\widetilde{\text{LOW}}$) izhodne mehke spremenljivke *Heat_power*, upoštevamo samo tisti sklep, katerega pravilo ima največjo stopnjo izpolnjenosti. Vrednost izhodne mehke spremenljivke zapišemo kot pripadnost z izrazom

$$\mu(u) = \max\{\mu_{12}^{\text{MEDIUM}}(u), \mu_{17}^{\text{LOW}}(u)\}. \quad (4.21)$$

Grafično predstavitev izraza (4.21) predstavlja slika 4.7.



Slika 4.7

Grafični prikaz združenih sklepov vseh mehkih pravil krmilnika za izhodno mehko spremenljivko *Heat_power* pri vrednosti vhodnih mehkih spremenljivk $\mu_{\epsilon}(22^{\circ}\text{C}) = 0.6$, $\mu_{\delta}(22^{\circ}\text{C}) = 0.2$ in Δ_{temp}
 $\mu_{\text{SS}}(-0.6^{\circ}\text{C}) = 0.57$, $\mu_{\text{Z}}(-0.6^{\circ}\text{C}) = 0.14$.

4.4.3 Ostrenje vrednosti mehkih spremenljivk

V realnem svetu pri krmiljenju ali opisovanju procesov operiramo z ostrimi numeričnimi vrednostmi, zato vrednost izhodnih mehkih spremenljivk pred izstopom iz mehkega sistema *ostrimo*. Poznamo več vrst metod ostrenja, izbira ustrezne metode pa je odvisna od narave mehkega odločitvenega sistema [50, 53]. Najpogosteje se za ostrenje mehkih vrednosti uporablja *težiščna metoda* (angl. *center of gravity* - COG ali *center of area* - COA), pri kateri ostro vrednost izračunamo po izrazu

$$O_{\text{COG}} = \frac{\int_u u \cdot \mu(u) du}{\int_u \mu(u) du}, \quad (4.22)$$

kjer je u središčnica pripadnostne funkcije $\mu(u)$ izhodne mehke spremenljivke iz izraza (4.21) in O_{COG} ostro vrednost izhodne spremenljivke. Pripadnostna funkcija $\mu(u)$ je včasih le delno zvezna ali celo diskretna. Ostro vrednost izhodne mehke spremenljivke v takšnih primerih izračunamo po izrazu

$$O_{\text{COG}} = \frac{\sum_{i=1}^n u_i \cdot \mu(u_i)}{\sum_{i=1}^n \mu(u_i)}. \quad (4.23)$$

Krmilnik za nastavljanje intenzivnosti ogrevanja uporablja ostro vrednost spremenljivke *Out*, ki smo jo definirali na intervalu $U = [0\%, 100\%]$. Ostrenje vrednosti

izhodne mehke spremenljivke *Heat_power*, ki smo jo pridobili z izrazom (4.21), izračunamo kot

$$Out_{COG} = \frac{0.2 \cdot 35 + 0.57 \cdot 65}{0.2 + 0.57} = 57.2. \quad (4.24)$$

Krmilnik bo torej pri trenutni ambientalni temperaturi $T_a(t) = 23\text{ }^{\circ}\text{C}$ in trenutni spremembi temperature $\Delta T_a(t) = -0.6\text{ }^{\circ}\text{C}$ nastavil intenzivnost ogrevanja $Out(t) = 57.2\text{ }\%$.

V pričujočem poglavju smo predstavili osnove teorije mehke logike in mehkih množic, na podlagi katerih smo definirali mehki krmilnik ogrevalnega sistema. Izhodna vrednost (intenzivnost ogrevanja) mehkega krmilnika lahko v odvisnosti od vrednosti vhodnih ostrih spremenljivk zavzame poljubno vrednost na intervalu od $[0\text{ }\%, 100\text{ }\%]$. V primeru, da mehke množice, ki smo jih uporabili pri določitvi vhodnih mehkih spremenljivk in izhodne mehke spremenljivke, zamenjamo z množicami iz klasične teorije množic, bi dobili le štiri možne izhodne nastavitve delovanja ogrevalnega sistema (ZERO, LOW, MEDIUM, HIGH), kar bi zmanjšalo natančnost krmiljenja. S povečevanjem števila ostrih množic, s katerimi predstavimo izhodne spremenljivke, bi natančnost povečali, a bi s tem bistveno povečali tudi procesno-računsko kompleksnost krmilnika.

Krmiljenje z mehko logiko nam omogoča, da krmiljeni proces predstavimo z mehki (lingvističnimi) spremenljivkami in mehki IF-THEN pravili, ki jih definiramo na podlagi izkustvenega poznavanja problema. Za izgradnjo mehkega odločitvenega sistema torej ne potrebujemo eksaktnega matematičnega modela, ki za dani problem ni na voljo, ali pa je ta težko določljiv.

*Uporaba mehke logike za
modeliranje dinamike
bioloških sistemov*

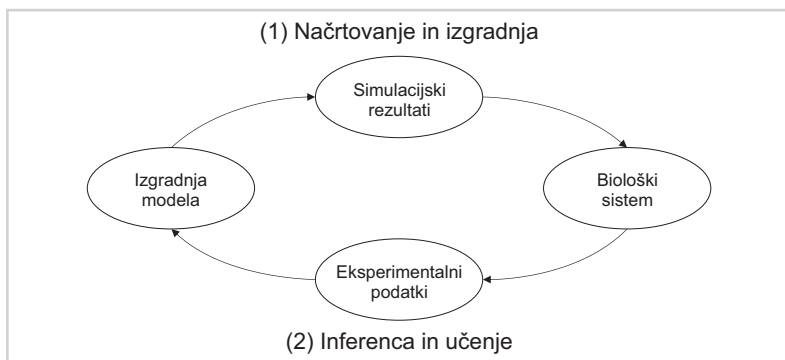
V pričujočem poglavju predstavimo možnosti uporabe mehke logike za modeliranje bioloških sistemov. Na področju modeliranja mehko logiko uporabljamo za reševanje naslednjih dveh vrst problemov:

1. za izgradnjo mehkega modela dinamike na osnovi približnega izkustvenega poznavanja sistema in
2. za ekstrakcijo znanja iz eksperimentalnih podatkov, s katerim odkrivamo posamezne medgenske interakcije, topologijo relacij celotnega gensko regulatornega omrežja ali zakonitosti neznane dinamike.

V prvi skupini so pristopi, s katerimi modeliramo dinamiko celotnih bioloških sistemov ali posameznih procesov v sistemu, pri katerih želimo posamezni proces oziroma kemijsko reakcijo, glede na približno izkustveno poznavanje zakonitosti dinamike in podatkov o lastnostih nastopajočih kemijskih zvrsti, ki jih imamo na voljo, opisati čim natančneje. V drugo skupino spadajo pristopi, pri katerih iz eksperimentalnih podatkov s postopki avtomatiziranega učenja iščemo interakcije v gensko regulatornem omrežju oziroma na podlagi eksperimentalnih podatkov zgradimo mehki sistem, ki opiše predhodno neznano dinamiko sistema. Problemski domeni, ki jih obravnavajo pristopi, sta prikazani na sliki 5.1.

Slika 5.1

Shema razvrstitve pristopov modeliranja glede na cilje modeliranja. Zgornji del sheme (načrtovanje in izgradnja) prikazuje problemsko domeno pristopov iz prve alineje, pri katerih na podlagi simulacijskih rezultatov modela biološkega sistema načrtujemo eksperimentalno delo, spodnji del sheme (uporaba eksperimentalnih podatkov) pa pristopov iz druge alineje, pri katerih iz eksperimentalnih podatkov zgradimo model, ki nam omogoča boljše razumevanje biološkega sistema.



V pričujočem poglavju najprej naredimo pregled literature celotnega področja modeliranja bioloških sistemov s pomočjo mehke logike. V nadaljevanju poglavja natančneje

predstavimo pristope iz prve točke in izpostavimo njihove pomanjkljivosti. Metodo, ki predstavlja izvirne doprinos doktorske disertacije in sodi v prvo skupino, predstavimo v poglavju 6.

5.1 Pregled literature

Mehka logika se za modeliranje bioloških sistemov v domeni gensko regulatornih omrežij uporablja za opis osnovnih kemijskih procesov. Ker so slednji relativno enostavni, jih lahko na lingvističen način opišemo z uporabo mehkih spremenljivk in mehkih IF-THEN pravil. Mehka logika se za izgradnjo modela biološkega sistema uporablja v sledeče namene:

- *izgradnja pol-kvantitativnega modela dinamike biološkega procesa na podlagi mehke logike in približnega izkustvenega poznavanja*: model zgradimo na podlagi približnega izkustvenega poznavanja procesov, kemijskih zvrsti, ki v procesih nastopajo, ter njihovih lastnosti in interakcij; tovrstni modeli so enostavni in uporabljajo mehka IF-THEN pravila, ki dinamiko procesa opisujejo pol-kvantitativno - opisujejo spremembe stanja sistema v času, pri čemer sprememb ali vrednosti spremenljivk (koncentracij opazovanih kemijskih zvrsti) ne moremo opisati z eksaktnimi merskimi enotami [17, 18, 54] (točka 1 v uvodu poglavja);
- *izgradnja kvantitativnega mehkega modela dinamike biološkega procesa na podlagi učenja iz eksperimentalnih podatkov*: s pomočjo avtomatiziranega postopka učenja iz eksperimentalnih podatkov pridobimo mehka IF-THEN pravila, ki posnemajo vpliv kinetičnih parametrov na dinamiko procesa; s pridobljenimi mehki pravili tako dobimo kvantitativni mehki lingvistični model dinamike biološkega procesa, ki je povsem primerljiv z ostalimi kvantitativnimi metodami modeliranja dinamike [55–58] (točka 2 v uvodu poglavja);
- *odkrivanje enostavnih relacij v topologiji gensko regulatornega omrežja na podlagi eksperimentalnih podatkov*: z analizo časovnih vrst koncentracij vseh opazovanih kemijskih zvrsti v sistemu zgradimo mehki model gensko regulatornega omrežja (topologijo relacij v omrežju), na podlagi katerega odkrijemo posamezne enostavne medgenske interakcije, ki jih opišemo kot interakcije med tremi tipi kemijskih zvrsti, in sicer aktivatorji, represorji in geni; zaradi ohranjanja manjše računske potratnosti se omejimo na interakcije med največ enim aktivatorjem,

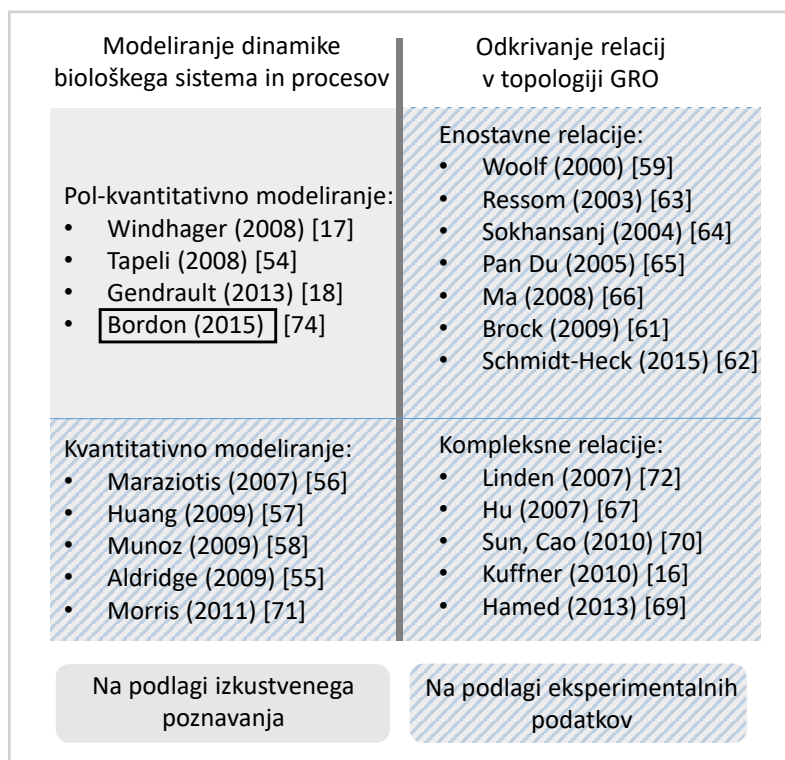
enim represorjem in enim opazovanim genom, pri čemer nas zanima le obstoj interakcije (gen je *aktiviran* ali *represiran*), ne pa tudi njena intenzivnost [59–66] (točka 2 v uvodu poglavja);

- *odkrivanje kompleksnih relacij v topologiji gensko regulatornega omrežja na podlagi eksperimentalnih podatkov*: z analizo časovnih vrst koncentracij vseh opazovanih kemijskih zvrsti v sistemu zgradimo mehki model gensko regulatornega omrežja (topologijo relacij v omrežju), ki zajema interakcije med posameznimi geni v gensko regulatornem omrežju in jih lingvistično opiše natančneje kot v predhodni alineji; za razliko od pristopov, opisanih v predhodni alineji, ki so računsko manj potratni, pri teh pristopih natančno raziščemo kompleksnejše interakcije, pri katerih na ekspresijo gena vplivata več kot dva transkripcijska faktorja (več represorjev in/ali aktivatorjev); z uporabo mehke logike v teh modelih poleg obstoja (gen je *aktiviran* ali *inhibiran*) na lingvističen način opišemo tudi intenzivnost interakcije (*šibka, srednja, močna* itd.), s čimer topologijo relacij in interakcije v gensko regulatornem omrežju model opiše natančneje [16, 19, 67–70] (točka 2 v uvodu poglavja).

Pristopi in njihovi avtorji, predstavljeni v predhodnih alinejah, so grafično razvrščeni glede na koncepte pristopov na sliki 5.2. V nadaljevanju izpostavimo značilnosti posameznih pristopov za modeliranje bioloških sistemov, ki temeljijo na mehki logiki. Pristopi opisujejo izgradnjo pol-kvantitativnih in kvantitativnih mehkih modelov bioloških procesov ter odkrivanje enostavnih in kompleksnih genskih interakcij v gensko regulatornih omrežjih iz časovnih vrst koncentracij opazovanih kemijskih zvrsti.

5.1.1 *Izgradnja pol-kvantitativnega modela dinamike biološkega procesa na podlagi mehke logike in približnega izkustvenega znanja*

Predpostavimo, da za opazovani biološki proces nimamo ne matematičnega modela ne eksperimentalnih podatkov, imamo pa približno izkustveno pridobljeno opisno poznavanje interakcij med aktivatorji, represorji in geni. V tem primeru lahko model posameznega biološkega procesa ali model dinamike celotnega sistema postavimo s pomočjo mehke logike. Posamezne biološke procese opišemo z mehкими odločitvenimi sistemi, s pomočjo katerih pridobimo pol-kvantitativni model in njemu ustrezne simulacijske rezultate. S tem se sicer včasih celo izognemo uporabi eksaktnih vrednosti



Slika 5.2

Mehka logika se pri modeliranju bioloških sistemov uporablja za pol-kvantitativno modeliranje dinamike bioloških procesov sistema (označeno z nečrtano površino - skupina 1) in kadar so na voljo eksperimentalni podatki za vzpostavitev kvantitativnega modela procesa ter pri modelih s katerimi odkrivamo interakcije v gensko regulatornih omrežjih iz eksperimentalnih podatkov (označeno s črtano površino - skupina 2).

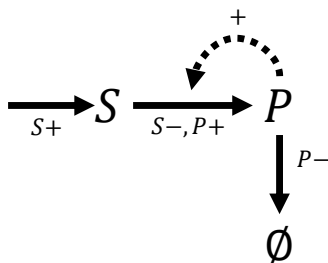
kinetičnih parametrov, a v večini primerov izgubimo stik z realnimi merskimi enotami biološkega procesa.

V delu [17], ki je domensko najbližje doprinosu pričujoče disertacije, avtorji na podlagi približnega izkustvenega poznavanja postavijo mehki model Higgins-Sel'kovega oscilatorja. Slednji sestoji iz dveh kemijskih zvrsti, substrata in produkta, ki nastopata v procesih nastajanja, pretvorbe in degradacije. Shemo oscilatorja prikazuje slika 5.3. V mehki model oscilatorja za koncentraciji substrata in produkta avtorji vpeljejo mehki spremenljivki z absolutnimi mehkiimi vrednostmi \overline{LOW} , \overline{MEDIUM} in \overline{HIGH} . Obe mehki spremenljivki v procesu mehkega odločanja nastopata kot vhodni in izhodni spremenljivki. Opis bioloških procesov, ki nastopajo v oscilatorju, in podrobnejšo

predstavitev mehkega modela podamo v razdelku 5.2. Drugih pristopov, ki bi brez eksperimentalnih podatkov z mehko logiko opisali dinamiko sprememb koncentracij kemijskih zvrsti v biološkem sistemu in bi spadali v ožje področje pričujoče disertacije, v literaturi ne najdemo. Iz omenjenega članka sicer izhaja veliko pristopov, ki opis z mehko logiko uporabljajo kot izhodišče za odkrivanje novih interakcij. Opišemo jih v razdelkih 5.I.3 in 5.I.4, kamor sodijo.

Slika 5.3

V mehkem modelu Higgins-Sel'kovega oscilatorja nastopata dve kemijski zvrsti, substrat (S) in produkt (P). Model opisuje procese nastajanja substrata, pretvorbe substrata v produkt in degradacijo produkta. Proces pretvorbe je ob prisotnosti produkta pospešen.



Ostali pristopi, ki uporabljajo mehko logiko za pol-kvantitativno modeliranje, se oddaljujejo od opisov dinamike posameznih procesov in mehko logiko uporabljajo zgolj kot razširitev Boolovih mrež, katerih uporabo smo podrobneje opisali v poglavju 3. Boolove mreže uporabljamo za preprost visokonivojski opis dinamike biološkega sistema. Slednje biološke procese v sistemu opišejo z dvema diskretnima vrednostma (npr. proces je *aktiven* ali *neaktiven*, vezava je *močna* ali *šibka*, aktivacija/inhibicija je *prisotna* ali *odsotna* itd.). Iz modela na osnovi Boolovih mrež lahko izpeljemo pol-kvantitativni model, za izgradnjo katerega potrebujemo zvezno predstavitev koncentracij opazovanih kemijskih zvrsti. V delu [18] avtorji z uporabo mehke logike v Boolovi mreži zvezno predstavijo koncentracije opazovanih kemijskih zvrsti, aktivnost gena in moč interakcije med geni. Dobljen mehki model, ki ga zgradijo na podlagi približnega izkustvenega poznavanja procesov in Boolovih mrež, uporabijo kot osnovo za vzpostavitev pol-kvantitativnega modela dinamike biološkega sistema.

Poznamo tudi pristope, ki mehko logiko uporabljajo za analizo ravnovesja pretoka (angl. *flux balance analysis*), pri kateri ločeno modeliramo gensko regulatorno in metabolno omrežje [54]. Slednja pri analizi ravnovesja pretoka običajno modeliramo s sistemom diferencialnih enačb, pri katerih za izračun novega stanja koncentracij meta-

bolitov v metabolnem delu uporabljamo hitrosti kemijskih reakcij, ki jih pridobimo z modeliranjem regulatornega dela. V primeru, da natančne vrednosti kinetičnih parametrov za kvantitativno modeliranje regulatornega dela niso na voljo, lahko ta del poenostavljeno modeliramo z mehko logiko. Kot izhodišče za modeliranje regulatornega dela v delu [54] avtorji uporabijo Boolove mreže, pri katerih posamezni gen označijo kot aktiven (izražanje) ali inhibiran (ni izražanja). Boolove mreže razširijo z mehko logiko in glede na trenutne koncentracije kemijskih zvrsti, ki vplivajo na dinamiko regulatornega dela, aktivnost gena določijo zvezno. Stopnjo izražanja nato prenesejo v metabolni del, v katerem jo uporabijo kot hitrost produkcije opazovanega metabolita.

5.1.2 Izgradnja kvantitativnega mehkega modela dinamike biološkega procesa na podlagi učenja iz eksperimentalnih podatkov

Za izgradnjo kvantitativnega mehkega modela dinamike biološkega procesa z obstoječimi pristopi, ki temeljijo na mehki logiki, potrebujemo eksperimentalne podatke, ki jih pridobimo z meritvami koncentracij skozi čas ali z merjenjem hitrosti vezave oziroma cepitve posameznih kemijskih zvrsti. Če za določen biološki sistem tovrstne eksperimentalne podatke imamo, lahko metode za ocenjevanje in pridobivanje kinetičnih parametrov kemijskih reakcij dopolnimo z mehko logiko.

Podobno kot v viru [18] avtorji vira [55] z mehko logiko razširijo Boolove mreže, pri katerih namesto prisotnosti oziroma odsotnosti kemijske zvrsti vrednosti koncentracij določijo zvezno. Za razliko od pristopa opisanega v viru [18], v katerem lastnosti mehkega modela določijo na podlagi približnega izkustvenega poznavanja procesov, avtorji pri tem pristopu lastnosti mehkih spremenljivk in mehka IF-THEN pravila določijo z avtomatiziranim postopkom učenja na podlagi eksperimentalnih podatkov. Z dobljenim modelom dinamiko posameznih bioloških procesov opišejo kvantitativno.

Pri modelih, ki temeljijo izključno na sistemu diferencialnih enačb, pogosto ne poznamo vrednosti vseh kinetičnih parametrov kemijskih reakcij, ki pogojujejo dinamiko sistema. V literaturi najdemo več pristopov, ki del modela, za katerega kinetičnih parametrov ne poznajo, nadomestijo s pristopom, ki uporablja mehko logiko. Za ta del modela uporabijo učilni postopek (npr. ANFIS - angl. *adaptive network-based fuzzy inference system*), ki na podlagi eksperimentalnih podatkov algoritmično določi mehka IF-THEN pravila in nabor mehkih spremenljivk ter njihove vrednosti, ki posnemajo vpliv vrednosti kinetičnih parametrov in omogočajo kvantitativno modeliranje z mehko logiko [56–58].

Pogosto nas zanima, kako se spremeni dinamika sistema, če določene dele sistema odstranimo. V viru [71] avtorji vplive dveh ali več kemijskih zvrsti združujejo z lingvističnimi pravili, s čimer navidezno odstranjujejo dele modela. S primerjanjem simulacijskih rezultatov novo pridobljenih manjših modelov nato določijo vpliv prisotnosti oziroma odsotnosti posameznih kemijskih zvrsti na dinamiko sistema.

5.1.3 Odkrivanje enostavnih relacij v topologiji gensko regulatornega omrežja na podlagi eksperimentalnih podatkov

Z razvojem metod za pridobivanje velikih količin podatkov o ekspresiji genov so se za analizo slednjih razvile različne metode. Za odkrivanje novih medgenskih interakcij v biološkem sistemu se je mehka logika uveljavila s pristopi, pri katerih z analizo časovnih vrst koncentracij mRNA iščemo obstoj enostavnih interakcij med posameznimi geni in s tem gradimo topologijo relacij gensko regulatornega omrežja, model dinamike sistema pa nas v tem primeru ne zanima. Pri enostavnih relacijah iščemo interakcije med tremi tipi zvrsti, in sicer aktivatorji, represorji in geni. Zaradi postopkov pridobivanja eksperimentalnih podatkov interakcij ne odkrivamo na podlagi opazovanja koncentracij transkripcijskih faktorjev, ampak opazujemo koncentracije mRNA, ki se pri translaciji prepisejo v aktivatorje oziroma represorje, ter koncentracijo mRNA, ki pripada genu, katerega izražanje regulira en aktivator in en represor. V mehkem modelu tako nastopata dve vhodni mehki spremenljivki (koncentraciji mRNA, ki se pri translaciji prepiseta v aktivator oziroma represor) in ena izhodna mehka spremenljivka (koncentracija mRNA, ki se povečuje zaradi izražanja opazovanega gena).

Pri pristopih, ki iščejo enostavne interakcije, avtorji v modelu predhodno določijo širši nabor enostavnih mehkih IF-THEN pravil z dvema vhodnima in eno izhodno mehko spremenljivko, s katerimi opišejo stanje sistema v vsaki časovni točki [59–62]. Iz pravil, ki ustrezno opišejo celotno časovno vrsto, sklepajo, kakšne so interakcije posameznih genov z ostalimi izbranimi geni. V postopku se preveri ustreznost vseh pravil za vsako točko v časovni vrsti, zato so ti pristopi računsko potratni.

Izboljšavo omenjenih pristopov predstavlja predhodno gručenje (angl. *clustering*) genov, ki izkazujejo podobno izražanje. Za vsako gručo genov se preverijo le tista mehka IF-THEN pravila, ki ustrezno opišejo izražanje v opazovani gručici, s čimer se bistveno zmanjša računsko potratnost odkrivanja novih medgenskih interakcij [63–65]. Predhodno gručenje uporabljajo tudi pri pristopih, s katerimi iščejo skupine genov, katerih izražanje regulira več skupnih transkripcijskih faktorjev (angl. *codependency*

regulation) [66].

5.1.4 Odkrivanje kompleksnih relacij v topologiji gensko regulatornega omrežja na podlagi eksperimentalnih podatkov

V metodah, ki smo jih opisali v predhodnem razdelku, se uporabljajo preprosta mehka IF-THEN pravila, ki vsebujejo natanko dve vhodni mehki spremenljivki in eno izhodno mehko spremenljivko ter opisujejo preprost princip genske regulacije, pri katerem je ekspresija gena zaradi transkripcijskih faktorjev bodisi aktivirana bodisi inhibirana.

Predpostavimo, da nas pri odkrivanju kompleksnih relacij v topologiji zanima le točno določen nabor različnih tipov interakcij. Majhno število različnih tipov interakcij avtorji vira [72] odkrivajo z določitvijo majhnega nabora mehkih IF-THEN pravil, ki opisujejo kompleksne interakcije. Pri slednjih poleg obstoja opisujejo tudi njihovo intenzivnost. Pravila določijo na podlagi obstoječega približnega izkustvenega poznavanja domene. Z uporabo manjšega števila pravil bistveno zmanjšajo računsko potratnost iskanja. V primeru, da iščemo veliko različnih tipov interakcij, lahko uporabimo pristope, pri katerih postavimo več manjših naborov mehkih IF-THEN pravil. Slednji posamezne tipe interakcij opišejo s stopnjo natančnosti, ki jo določimo hevristično. Na podlagi različnih tipov hevristike ohranimo ustrezne ali ovržemo neustrezne nabo-re pravil ter tako odkrijemo morebitne nove interakcije [19, 67–69].

V viru [16] avtorji poleg različnih tipov interakcij iščejo različne tipe transkripcijskih faktorjev, ki jih ločijo po moči (*močni*, *srednje močni*, *šibki*) in tako vpliv transkripcijskih faktorjev v omrežju definirajo natančneje. Ker se računska potratnost iskanja zaradi veliko večjega števila možnih tipov povezav bistveno poveča, najboljšo različico celotnega modela iščejo s pomočjo genetskega algoritma, ki išče mehka IF-THEN pravila in določa tipe transkripcijskih faktorjev.

V primeru, da iščemo in opisujemo vse obstoječe tipe medgenskih interakcij v velikih gensko regulatornih omrežjih, problem hitro postane neobvladljiv. V viru [70] ta problem rešujejo tako, da na podlagi približnega izkustvenega poznavanja sistema določijo manjši nabor za sistem specifičnih interakcij. Tipe interakcij, ki se v opazovanem sistemu ne pojavljajo, izključijo iz iskalnega postopka in s tem bistveno zmanjšajo računsko potratnost postopka. Tovrstni modeli so zelo natančni, a jih je zaradi kompleksnosti težko interpretirati, saj v dobljenih pravilih nastopa veliko vhodnih mehkih spremenljivk.

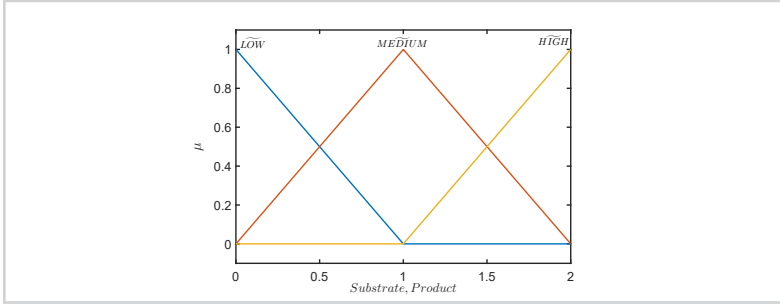
5.2 Mehki pol-kvantitativni model Higgins-Sel'kovega oscilatorja

V pričujočem razdelku predstavimo mehki pol-kvantitativni model dinamike Higgins-Sel'kovega oscilatorja, katerega osnovni opis smo podali v razdelku 5.1.1 in služi kot primer uporabe obstoječega pristopa za modeliranje bioloških sistemov z mehko logiko, ki je najbližje doprinosu pričujoče disertacije. Primerjamo simulacijske rezultate mehkega modela Higgins-Sel'kovega oscilatorja, povzetega po [17], ter modela, ki temelji na sistemu diferencialnih enačb, povzetem po [73]. Cilj primerjave modelov je prikazati pomanjkljivost mehkega modela, pri katerem ne obravnavamo relativnih sprememb koncentracij. Mehki pol-kvantitativni model so avtorji definirali na podlagi lingvističnega opisa približnega izkustvenega poznavanja dinamike oscilatorja, ki ga lahko povzamemo v naslednjih alinejah:

- model oscilatorja opisuje proces pretvorbe, v katerem nastopata dve kemijski zvrsti, substrat in produkt,
- produkcija substrata je konstantna,
- v procesu pretvorbe substrata v produkt se koncentracija produkta povečuje, koncentracija substrata pa zmanjšuje,
- proces pretvorbe substrata v produkt je pri višji koncentraciji produkta hitrejši,
- koncentracija produkta se zaradi procesa degradacije zmanjšuje.

V mehkem modelu, ki smo ga povzeli neposredno po viru [17], nastopata mehki spremenljivki *Substrate* in *Product*, ki predstavljata velikosti koncentracij obeh kemijskih zvrsti. Definirani sta na intervalu ostrih vrednosti od 0 do 2 ($U = [0, 2]$) brez merskih enot. Na intervalu U so avtorji za obe mehki spremenljivki definirali nabor treh mehkih množic $T = \{\widetilde{\text{LOW}}, \widetilde{\text{MEDIUM}}, \widetilde{\text{HIGH}}\}$, ki jih opisujejo enakomerno razporejene trikotne pripadnostne funkcije, prikazane na sliki 5.4.

Lingvistični opis dinamike avtorji razdelijo na opis absolutnih sprememb koncentracij produkta in opis absolutnih sprememb koncentracij substrata. Predstavijo ju z naslednjima naboroma mehkih IF-THEN pravil:



Slika 5.4

Definicija mehkih množic in njihovih pripadnostnih funkcij mehkih spremenljivk *Substrate* in *Product*, ki v mehkem modelu nastopata kot vhodna in izhodna mehka spremenljivka.

$$\begin{aligned}
 r_{P1}: & \text{ IF (Substrate is } \widetilde{\text{LOW}} \text{) THEN (Product is } \widetilde{\text{LOW}} \text{),} \\
 r_{P2}: & \text{ IF (Substrate is NOT } \widetilde{\text{LOW}} \text{) AND (Product is } \widetilde{\text{LOW}} \text{)} \\
 & \text{ THEN (Product is } \widetilde{\text{MEDIUM}} \text{),} \\
 r_{P2}: & \text{ IF (Substrate is NOT } \widetilde{\text{LOW}} \text{) AND (Product is NOT } \widetilde{\text{LOW}} \text{)} \\
 & \text{ THEN (Product is } \widetilde{\text{HIGH}} \text{).} \quad (5.1)
 \end{aligned}$$

$$\begin{aligned}
 r_{S1}: & \text{ IF (Product is } \widetilde{\text{HIGH}} \text{) THEN (Substrate is } \widetilde{\text{LOW}} \text{),} \\
 r_{S2}: & \text{ IF (Substrate is } \widetilde{\text{LOW}} \text{) AND (Product is NOT } \widetilde{\text{HIGH}} \text{)} \\
 & \text{ THEN (Substrate is } \widetilde{\text{MEDIUM}} \text{),} \\
 r_{S3}: & \text{ IF (Substrate is NOT } \widetilde{\text{LOW}} \text{) AND (Product is NOT } \widetilde{\text{HIGH}} \text{)} \\
 & \text{ THEN (Substrate is } \widetilde{\text{HIGH}} \text{).} \quad (5.2)
 \end{aligned}$$

Pravila v naboru 5.2 konstantne produkcije substrata ne podajajo eksplicitno, ampak implicitno v povezavi s koncentracijo produkta. V primeru visoke koncentracije produkta se predvideva, da je zaradi porabe substrata pri pretvorbi koncentracija slednjega trenutno nizka. V primeru, da koncentracija produkta ni visoka, se substrat za pretvorbo predvidoma ni porabil in se posledično njegova koncentracija poviša iz nizke na srednjo raven in nato iz srednje na visoko raven.

Simulacijske rezultate mehkega modela so avtorji primerjali s simulacijskimi rezultati modela na osnovi sistema diferencialnih enačb, ki so ga povzeli po [73]. Dinamiko

slednjega opisuje naslednji sistem diferencialnih enačb:

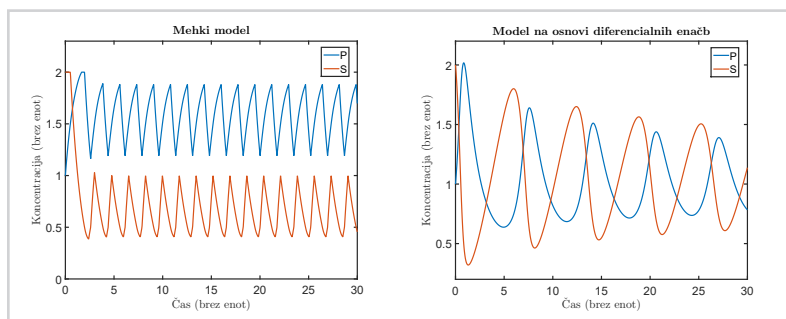
$$\frac{dS}{dt} = v_0 - k_1 SP^2, \quad (5.3)$$

$$\frac{dP}{dt} = k_1 SP^2 - k_2 P, \quad (5.4)$$

kjer sta S in P trenutni koncentraciji substrata in produkta, v_0 hitrost produkcije substrata, k_1 hitrost pretvorbe substrata v produkt in k_2 hitrost degradacije produkta. Oba modela smo realizirali v programskem orodju MATLAB Simulink® [51]. Pri obeh modelih izražanje časa in koncentracije ni predstavljeno s konkretnimi merskimi enotami, temveč z abstraktnimi enotami. Kot začetni koncentraciji substrata in produkta smo za potrebe simulacije predpostavili vrednosti $S = 2$ in $P = 1$ enot. Dolžina simulacije je 30 abstraktnih časovnih enot, pri čemer smo simulirali 1000 diskretnih časovnih korakov v eni abstraktni časovni enoti. Pri obeh modelih smo v simulacijskem okolju za numerično integriranje uporabili metodo *ode4* (Runge-Kutta). Simulacijski rezultati obeh modelov so prikazani na sliki 5.5.

Slika 5.5

Primerjava rezultatov mehkega modela (a) in modela na osnovi sistema diferencialnih enačb (b). Simulacijski rezultati mehkega modela sicer odražajo oscilacije, a se te bistveno razlikujejo od oscilacij v rezultatih modela na osnovi sistema diferencialnih enačb.



V mehkem modelu novo stanje sistema določimo absolutno glede na trenutno stanje sistema, medtem ko v modelu na osnovi sistema diferencialnih enačb novo stanje sistema določimo z majhnimi spremembami, ki jih v vsakem koraku simulacije izračunamo tako na podlagi trenutnega stanja kot tudi vrednosti kinetičnih parametrov, ki pogujejo dinamiko. Zaradi različne obravnave spremembe stanja sistema, ki je predstavljena absolutno v primeru mehkega modela in z majhnimi spremembami (relativno) v primeru modela na osnovi diferencialnih enačb, simulacijski rezultati prikazujejo povsem

drugačno dinamiko. Mehki model, ki spremembe stanja opisuje absolutno, torej ni primeren za kvantitativni opis dinamike sistema.

Simulacijski rezultati mehkega modela oscilatorja, ki smo ga predstavili v tem razdelku, se nekoliko razlikujejo od simulacijskih rezultatov mehkega modela, ki je predstavljen v viru [17]. Rezultati obeh mehkih modelov se razlikujejo predvsem v frekvenci in amplitudi oscilacij. Natančnega postopka pridobivanja simulacijskih rezultatov za mehki model v omenjenem viru avtorji ne opišejo. Nastanek razlike v rezultatih lahko deloma pripišemo razliki v načinu simuliranja mehkega modela, glede na to, da so tako definicije mehkih spremenljivk kot tudi nabora mehkih IF-THEN pravil enaki. Kljub tej razliki pa pri simulacijskih rezultatih obeh mehkih modelov opazimo podobnosti, in sicer da se ti precej razlikujejo od rezultatov modela na osnovi sistema diferencialnih enačb. Rezultati obeh mehkih modelov izkazujejo stabilne oscilacije, medtem ko so pri rezultatih modela na osnovi sistema diferencialnih enačb oscilacije dušene. Poleg tega sta frekvenca in amplituda oscilacij v rezultatih obeh mehkih modelov različni od frekvenca in amplitude oscilacij v rezultatih modela na osnovi sistema diferencialnih enačb.

Pomanjkljivost mehkega modela oscilatorja je predvsem v predstavitvi spremembe stanja. Mehki model spremembo stanja opisuje absolutno. Iz modela na osnovi sistema diferencialnih enačb je razvidno, da je odvisnost hitrosti nastajanja produkta pri pretvorbi od koncentracije produkta kvadratna in ne linearna, česar z absolutno predstavitvijo spremembe stanja ne moremo opisati natančno. Definiciji mehkih spremenljivk *Substrate* in *Product* sta enostavni, kar dodatno zmanjšuje natančnost. Za obe mehki spremenljivki, ki nastopata kot vhodni in izhodni spremenljivki, namreč uporabljamo le eno definicijo, s čimer težko opišemo kompleksnejše zakonitosti dinamike (npr. kvadratno odvisnost). Natančnost izgubimo tudi z definicijo naborov mehkih IF-THEN pravil, pri katerih nastajanje substrata opišemo implicitno s spreminjanjem koncentracije produkta.

Za povečanje natančnosti simulacijskih rezultatov mehkega modela bi morali spremembe stanja predstaviti drugače, z definicijo mehkih spremenljivk in mehkih IF-THEN pravil pa bolj natančno zajeti zakonitosti dinamike procesa, ki ga opisujemo.



*Modeliranje dinamike
enostavnih bioloških procesov z
neznanimi vrednostmi
kinetičnih parametrov*

6

V pričujočem poglavju predstavimo nov pristop modeliranja bioloških sistemov, ki predstavlja izvirni znanstveni doprinos doktorske disertacije [74]. Z opisanim pristopom modeliramo biološke sisteme v nastajanju ob odsotnosti natančnih vrednosti vseh ali le določenih kinetičnih parametrov, ki pogojujejo dinamiko biološkega sistema. Model opazovanega biološkega procesa, za katerega vrednosti kinetičnih parametrov niso znane, z našim pristopom zgradimo na podlagi mehke logike.

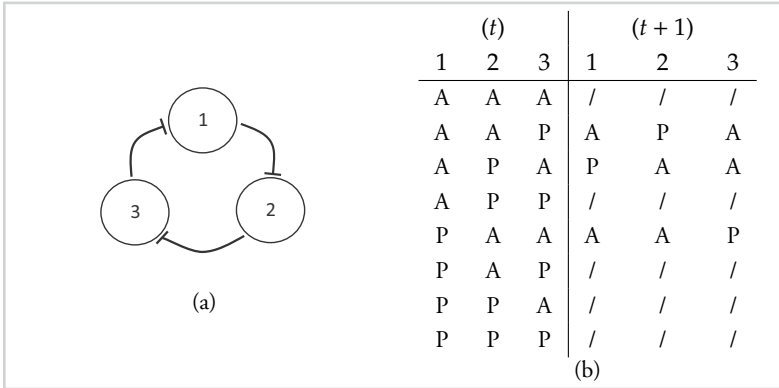
Predstavimo postopek izgradnje modela biološkega procesa z manjkajočimi vrednostmi kinetičnih parametrov ter postopek združitve mehkega modela procesa z modelom, ki temelji na podlagi sistema diferencialnih enačb. Postopek izgradnje mehkega modela predstavimo na primeru procesa transkripcije v represilatorju s tremi členi. Simulacijske rezultate dobljenega modela primerjamo z referenčnim modelom, zgrajenim izključno na podlagi sistema diferencialnih enačb.

6.1 Model genetskega represilatorja

Represilator je dinamičen biološki sistem, ki temelji na ekspresiji ali izražanju več genov v sistemu. Produkt posameznega gena zavira ekspresijo enega od preostalih genov. Vplivi produktov na ekspresije posameznih genov so povezani tako, da dobimo krožno represijski shemo. Primer represilatorja s tremi členi prikazuje slika 6.1a, njegovo delovanje pa opisuje tabela 6.1b. Dinamika represilatorja temelji na izražanju poljubnega števila genov. Iz literature je znano, da z determinističnim modeliranjem stabilnih oscilacij pri represilatorju s sodim številom genov v krožni represijski shemi ne dobimo, pri represilatorju z lihim številom genov v shemi pa samo pri določenih vrednostih kinetičnih parametrov modela, ki pogojujejo dinamiko sistema [8].

Dinamiko represilatorja določajo procesi produkcije vseh nastopajočih tipov mRNA (transkripcija), produkcije vseh nastopajočih tipov pripadajočih proteinov (translacija) in degradacije vseh nastopajočih tipov mRNA in proteinov. Transkripcija $mRNA_i$ i -tega izhodnega proteina je zaradi represije ustavljena ob prisotnosti proteina P_j , pri čemer indeks j določa indeks proteina, ki inhibira produkcijo $mRNA_i$ in velja $j = i - 1 \pmod{G}$, kjer je G število genov v represilatorju. Za primer represilatorja opisanega v pričujočem poglavju je $G = 3$. Intenzivnost posameznega procesa transkripcije je torej odvisna od prisotnosti ali koncentracije proteina, ki nastopa v vlogi represorja, intenzivnost posameznega procesa translacije od prisotnosti ali koncentracije mRNA, posamezni proces degradacije pa od prisotnosti ali koncentracije mRNA in posameznega proteina. Hitrosti navedenih procesov pogojujejo vrednosti kinetičnih parametrov

Slika 6.1



Primer sheme represilatorja s tremi členi (a), katerega delovanje opisuje tabela (b). V tabeli so na levi strani prikazane kombinacije stanj prisotnosti posameznega proteina, ki zavira ekspresijo enega od genov v shemi, v časovni točki t in na desni stanja koncentracij posameznega proteina v naslednji časovni točki $t+1$ (A - absent, P - present, / - neveljavna vhodna kombinacija). Ureditev genov v krožno represijsko shemo povzroči, da pri določenih vrednostih kinetičnih parametrov represilator stabilno oscilira.

posameznega procesa. Za potrebe predstavitve pristopa izgradnje mehkega modela predpostavimo, da vrednosti kinetičnih parametrov, ki pogojujejo hitrosti transkripcij vseh treh mRNA, niso znane.

6.2 Referenčni deterministični model represilatorja s tremi členi

Izhodiščni referenčni model represilatorja s tremi členi temelji na sistemu navadnih diferencialnih enačb (angl. *ordinary differential equation* - ODE). Predpostavimo, da imajo vsi členi represilatorja enake kinetične parametre. Za opis procesov uporabimo izraza [7, 8]

$$\frac{d[P_i(t)]}{dt} = k_{tsl} \cdot [mRNA_i(t)] - k_{degP} \cdot [P_i(t)], \quad i = 1, 2, 3, \quad (6.1)$$

$$\frac{d[mRNA_i(t)]}{dt} = \frac{k_{tsk}}{1 + [P_j(t)]^n} - k_{degmRNA} \cdot [mRNA_i(t)], \quad (6.2)$$

$$i = 1, 2, 3; \quad j = i - 1,$$

kjer izraz (6.1) predstavlja spremembo koncentracije i -tega proteina na osnovi procesov translacije in degradacije, v katerem je $[P_i]$ trenutna koncentracija i -tega proteina, $[mRNA_i]$ trenutna koncentracija pripadajoče i -te mRNA, k_{tsl} hitrost translacije in k_{degP} hitrost degradacije proteina, izraz (6.2) pa predstavlja spremembo koncentracije

mRNA na osnovi procesov transkripcije in degradacije, v katerem je $[P_i]$ trenutna koncentracija represorja, ki inhibira produkcijo i -te mRNA, $[mRNA_i]$ trenutna koncentracija i -te mRNA, n Hillov koeficient, k_{tsk} pa hitrost transkripcije. Tabela 6.1 prikazuje vrednosti kinetičnih parametrov, ki smo jih za potrebe pridobivanja simulacijskih rezultatov povzeli po viru [8]. Model represilatorja smo zgradili v okolju MATLAB Simulink® [51]. Diagram modela prikazuje slika 6.2. Na njej je z rdečo obkrožen proces transkripcije, z modro proces degradacije mRNA, z zeleno proces translacije in z rumeno proces degradacije proteinov.

Tabela 6.1

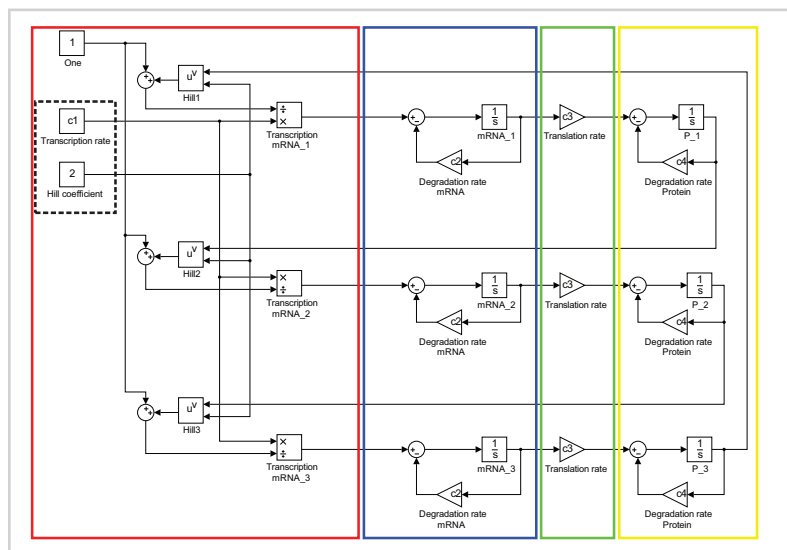
Vrednosti kinetičnih parametrov, ki smo jih uporabili v izhodiščnem referenčnem modelu [8].

<i>n.</i>	<i>Process</i>	<i>Parameter</i>	<i>Vrednost</i>
(1)	Transkripcija	k_{tsk}	1.6 min^{-1}
(2)	Transkripcija	n	2
(3)	Translacija	k_{tsl}	2.6 min^{-1}
(4)	Degradacija mRNA	$k_{deg_{mRNA}}$	0.12 min^{-1}
(5)	Degradacija proteina	k_{deg_p}	0.06 min^{-1}

Če vrednosti kateregakoli od kinetičnih parametrov, ki pogojujejo dinamiko sistema, niso znane, numerična simulacija modela na sliki 6.2 ni mogoča. V primeru, da vrednosti kinetičnih parametrov, ki sta na sliki 6.2 označena s črtkano črto, nista znani, prispevkov iz blokov Transcription mRNA_ i , $i = 1, 2, 3$ ni mogoče izračunati, saj imajo vhode, ki jih ni mogoče ovrednotiti.

6.3 Postopek izgradnje mehkega modela biološkega procesa z manjkajočimi vrednostmi parametrov

Za potrebe izgradnje mehkega modela opazovanega biološkega procesa najprej identificiramo znane in neznane vrednosti kinetičnih parametrov. Sledi analiza odvisnosti dinamike opazovanega procesa od različnih vrednosti sicer neznanih kinetičnih parametrov. Na podlagi analize odvisnosti definiramo osnovne značilnosti procesa ter jih opišemo lingvistično. S pomočjo lingvističnega opisa procesa nato definiramo mehke spremenljivke in določimo mehka IF-THEN pravila. Izgradnjo mehkega modela biološkega procesa tako razdelimo v štiri ključne korake:



Slika 6.2

Model represitorja s tremi členi v okolju MATLAB Simulink [51]. Z rdečo je obkrožen proces transkripcije, z modro proces degradacije mRNA, z zeleno proces translacije in z rumeno proces degradacije proteinov, s črtkano črto pa sta obkrožena hitrost transkripcije in Hillov koeficient.

1. identifikacija spremenljivk ter razpoložljivih in manjkajočih kinetičnih parametrov opazovanega procesa,
2. analiza odvisnosti opazovanega procesa od vrednosti sicer manjkajočih kinetičnih parametrov,
3. identifikacija lingvističnega opisa dinamike opazovanega procesa,
4. izgradnja mehkega modela, ki sestoji iz mehkih IF-THEN pravil in definicije mehkih spremenljivk.

Na vходу in izhodu dobljenega mehkega modela opazovanega procesa zaradi potrebe povezovanja spremenljivke nastopajo v ostri obliki. Za potrebe mehkega odločanja jih na vходу mehkega sistema predhodno mehčamo, na izhodu mehkega sistema pa ostrimo. Spremenljivke mehkega modela na vходу predstavljajo trenutne ostre vrednosti koncentracij kemijskih vrst, ki nastopajo v opazovanem procesu, medtem ko spremenljivke na izhodu mehkega modela predstavljajo ostre vrednosti sprememb koncentracij kemijskih vrst, na katere proces vpliva. Ostro vrednost spremembe koncen-

tracije, ki je lahko pozitivna ali negativna, uporabimo za izračun novega stanja koncentracije. Takšna definicija vhodnih in izhodnih spremenljivk nam omogoča, da mehki model združimo z modelom, ki je zgrajen na podlagi sistema diferencialnih enačb in ki za pridobitev simulacijskih rezultatov potrebuje vrednosti tako vhodnih kot tudi izhodnih spremenljivk v ostri obliki.

6.4 Izgradnja mehkega modela represije transkripcije

V izrazu (6.2) smo predstavili izračun transkripcije in degradacije mRNA. V pričujočem razdelku predpostavimo, da vrednosti kinetičnih parametrov k_{tsk} , ki predstavlja hitrost transkripcije, in n , ki predstavlja Hillov koeficient, nista znani. Slednja sta v shemi na sliki 6.2 označena s črtkano obrobo v rdečem obrobljenem delu. Parametra nastopata v prvem delu izraza (6.2) in vplivata le na hitrost procesa transkripcije. Ostale parametre povzamemo po tabeli 6.1. Pri izgradnji mehkega modela procesa transkripcije sledimo korakom, ki smo jih opisali v razdelku 6.3.

6.4.1 Identifikacija spremenljivk ter razpoložljivih in manjkajočih parametrov opazovanega procesa

Proces transkripcije opisuje prvi del izraza (6.2)

$$\frac{d[mRNA_i(t)]}{dt} = \frac{k_{tsk}}{1 + [P_j(t)]^n}, \quad (6.3)$$

ki sestoji iz naslednjih spremenljivk in parametrov:

- $[P_j]$: trenutna koncentracija represorja ($j = 1, 2, 3$), ki v našem mehkem modelu predstavlja vhodno spremenljivko, katero na vходу mehkega sistema procesa transkripcije iz ostre pretvorimo v mehko,
- $d[mRNA_i]$: sprememba koncentracije $mRNA_i$ ($i = 1, 2, 3$), ki v mehkem modelu nastopa kot izhodna spremenljivka; na izhodu mehkega sistema jo ostrimo; pri izračunu preostalega dela izraza (6.2) nastopa v ostri obliki,
- n : Hillov koeficient; predpostavimo, da njegova natančna vrednost ni znana; vpliv Hillovega koeficienta na dinamiko procesa določimo na podlagi analize odvisnosti opisane v razdelku 6.4.2,

- k_{tsk} : hitrost transkripcije; predpostavimo, da njena natančna vrednost ni znana; vpliv hitrosti transkripcije na dinamiko procesa določimo na podlagi analize odvisnosti opisane v razdelku 6.4.2.

Prispevek procesa transkripcije v izrazu (6.2) zamenjamo s prispevkom hipotetičnega mehkega modela transkripcije v obliki spremenljivke $FL_{Transcription}([P_j(t)])$. Tako dobimo izraz

$$\frac{d[mRNA_i(t)]}{dt} = FL_{Transcription}([P_j(t)]) - k_{deg_{mRNA}} \cdot [mRNA_i(t)], \quad (6.4)$$

kjer je $FL_{Transcription}([P_j(t)])$ ostra spremenljivka, ki predstavlja spremembo koncentracije izračunano na podlagi mehkega modela transkripcije.

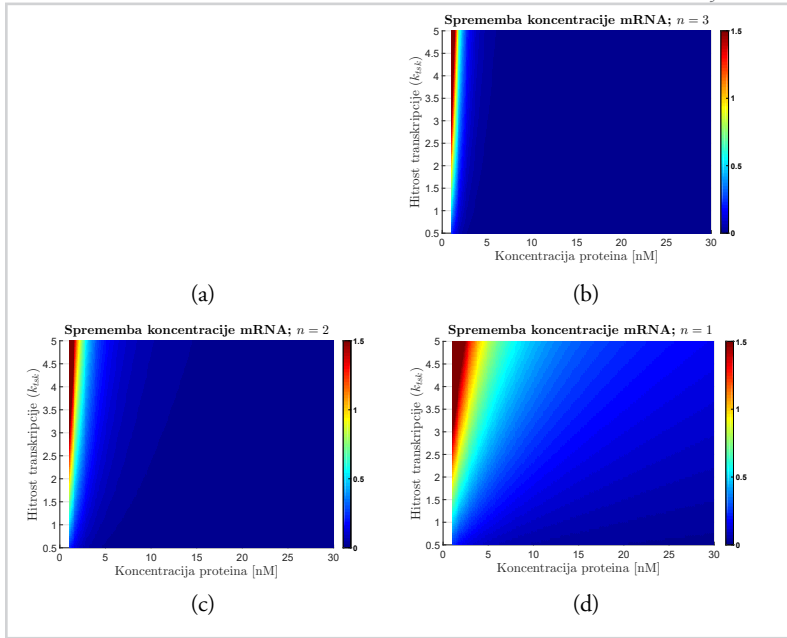
6.4.2 Analiza odvisnosti procesa transkripcije od vrednosti parametrov

Za potrebe izgradnje modela predhodno analiziramo, na kakšen način različni nabori vrednosti parametrov k_{tsk} (hitrost transkripcije) in n (Hillov koeficient) vplivajo na spremembo koncentracije $mRNA_i$ pri danih vrednostih koncentracije proteina P_j . Vrednosti parametrov, ki jih povzamemo po tabeli 6.1, so po predpostavki za vse tri člene enake. Na osnovi izraza (6.3) smo odvisnost preverili pri več biološko relevantnih vrednostih parametrov n in k_{tsk} . Na sliki 6.3 je prikazana odvisnost koncentracije mRNA od vrednosti parametra k_{tsk} , pri katerem smo vrednost Hillovega koeficienta fiksirali na $n = 1, 2, 3, 4$, medtem ko slika 6.4 prikazuje odvisnost koncentracije mRNA od vrednosti n , pri katerem smo vrednost parametra k_{tsk} fiksirali na $k_{tsk} = 0.5, 1, 3, 10 \text{ min}^{-1}$.

6.4.3 Identifikacija lingvističnega opisa dinamike procesa

Na slikah 6.3 in 6.4 zaradi preglednosti rezultatov prikazujemo samo zelo nizke koncentracije represorja. Na podlagi na slikah predstavljene dinamike lahko proces transkripcije opišemo z naslednjim naborom lingvističnih pravil:

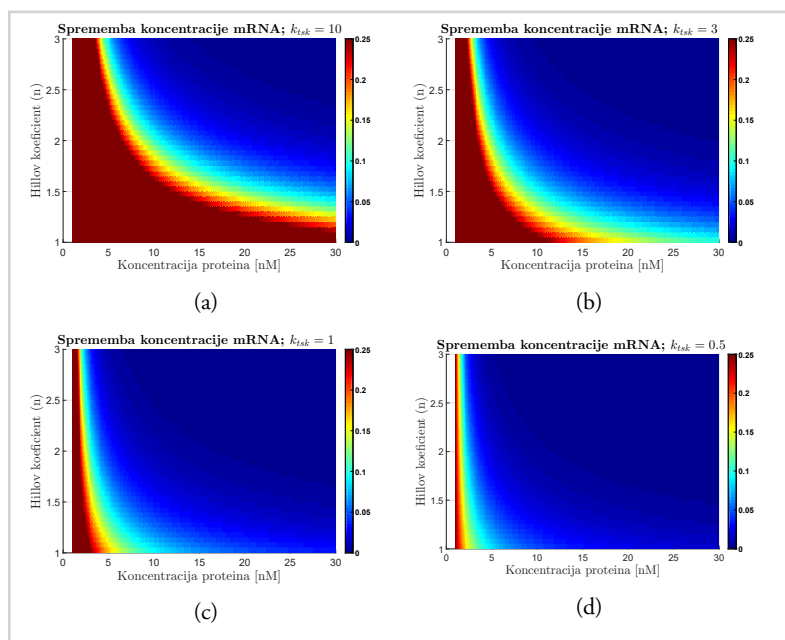
- *Interval koncentracije represorja lahko razdelimo na dva dela: ozek interval nizke koncentracije, v katerem transkripcija poteka* (npr. do približno 15 nM pri $k_{tsk} = 5 \text{ min}^{-1}$ na sliki 6.3d) *in preostali del intervala, v katerem je transkripcija povsem ustavljena* (npr. za koncentracije višje od 15 nM pri $k_{tsk} = 5 \text{ min}^{-1}$ na sliki 6.3d).



Slika 6.3

Velikost spremembe koncentracije mRNA. Grafi prikazujejo odvisnost spremembe koncentracije mRNA glede na parameter k_{tsk} pri vrednostih Hillovega koeficienta $n = 4$ (a), $n = 3$ (b), $n = 2$ (c) in $n = 1$ (d).

- *Transkripcija poteka samo v intervalu nizke koncentracije represorja* (kljub prikazanemu samo zelo ozkemu intervalu koncentracije represorja $[0, 30 \text{ nM}]$, že znotraj tega intervala povišanje koncentracije mRNA pri večini grafov pade na nič, npr. na vseh slikah 6.3 ter na slikah 6.4b, 6.4c in 6.4d).
- *Znotraj intervala nizke koncentracije represorja se sprememba koncentracije mRNA povečuje z nižanjem koncentracije represorja* (npr. na sliki 6.3d pri $k_{tsk} = 5 \text{ min}^{-1}$ od koncentracije represorja 15 nM do 0 nM sprememba koncentracije mRNA narašča).
- *Z večanjem parametra k_{tsk} se sprememba koncentracije mRNA veča linearno* (na sliki 6.3 se pri vseh štirih grafih vidi, da se pas neničelne spremembe koncentracije mRNA linearno širi vzdolž osi y).
- *Z nižanjem parametra n se interval na katerem transkripcija poteka razširi* (npr. na sliki 6.4c je pri $n = 3$ ta interval $[0, 5 \text{ nM}]$, pri $n = 1$ pa $[0, 20 \text{ nM}]$).



Slika 6.4

Velikost spremembe koncentracije mRNA. Grafi prikazujejo odvisnost spremembe koncentracije mRNA glede na vrednost Hillovega koeficienta pri hitrostih transkripcije $k_{lisk} = 10 \text{ min}^{-1}$ (a), $k_{lisk} = 3 \text{ min}^{-1}$ (b), $k_{lisk} = 1 \text{ min}^{-1}$ (c) in $k_{lisk} = 0.5 \text{ min}^{-1}$ (d).

Na podlagi teh ugotovitev identificiramo in definiramo mehke spremenljivke, določimo nabor mehkih IF-THEN pravil ter zgradimo mehki model transkripcije.

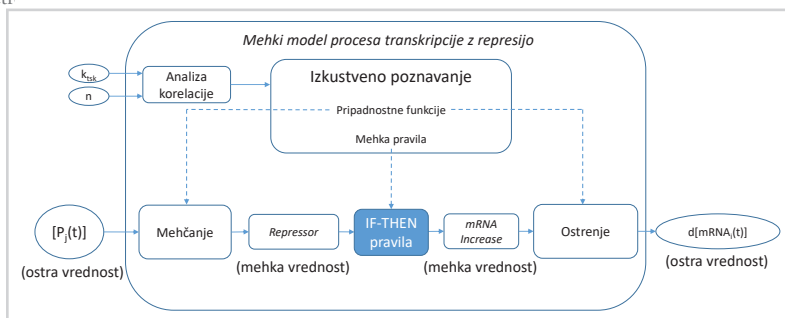
6.4.4 Izgradnja mehkega modela

Mehki model, ki ga bomo zgradili na podlagi lingvističnega opisa in analize parametrov iz prejšnjih razdelkov, prikazuje slika 6.5.

Mehko vhodno spremenljivko, ki predstavlja koncentracijo represorja, poimenujemo *Repressor* in za opis njenih vrednosti na podlagi prve in druge alineje lingvističnega opisa v razdelku 6.4.3 definiramo nabor mehkih množic $T = \{\overline{\text{ABSENT}}, \overline{\text{PRESENT}}\}$, kjer mehka množica $\overline{\text{ABSENT}}$ pokriva del intervala, na katerem transkripcija zaradi odsotnosti represorja poteka, mehka množica $\overline{\text{PRESENT}}$ pa preostali del intervala, kjer je transkripcija ustavljena. Mehko izhodno spremenljivko, ki predstavlja spremembo koncentracije mRNA, poimenujemo *mRNAIncrease*. Njena vrednost se povečuje, kadar represor ni prisoten ali če se njegova koncentracija zniža na zelo nizko vrednost.

Slika 6.5

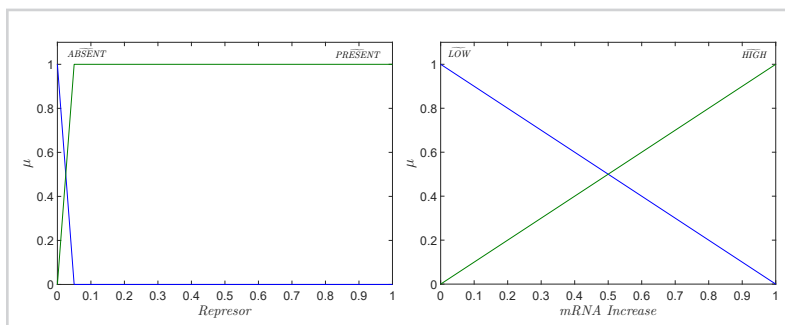
V sistem kot vhodna spremenljivka vstopa ostra vrednost koncentracije proteina P_j , izstopa pa povečanje koncentracije $mRNA_i$. Vrednosti izhodne spremenljivke določimo glede na lingvistični opis, ki smo ga formulirali na podlagi analize odvisnosti.



Ker bo v primeru nižanja koncentracije represorja znotraj intervala $\overline{\text{ABSENT}}$ sprememba koncentracije mRNA naraščala hitro, za njen opis definiramo nabor mehkih množic $T = \{\overline{\text{LOW}}, \overline{\text{HIGH}}\}$, kjer $\overline{\text{LOW}}$ opisuje ustavljeno oziroma zelo počasno transkripcijo in $\overline{\text{HIGH}}$ hitro transkripcijo, kadar je koncentracija represorja blizu 0 in bo posledično spreminjanje koncentracije mRNA največje. Slika 6.6 prikazuje oblike pripadnostnih funkcij mehkih množic vhodne mehke spremenljivke *Repressor* in izhodne mehke spremenljivke *mRNAIncrease*, ki smo jih določili na podlagi lingvističnega opisa procesa v razdelku 6.4.3.

Slika 6.6

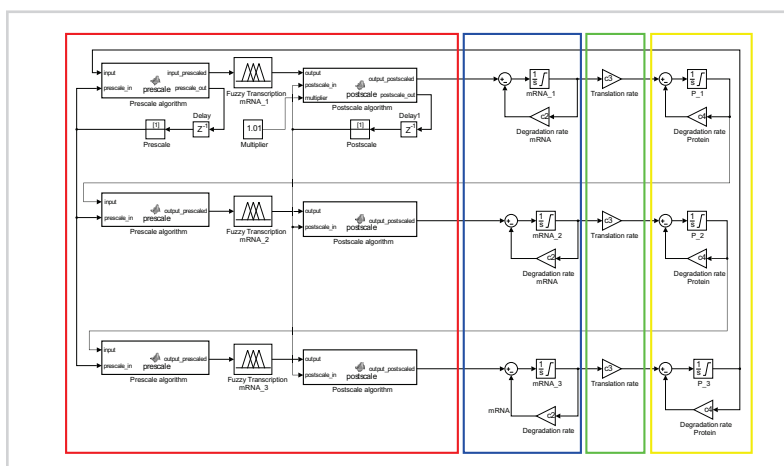
Pripadnostne funkcije mehkih množic, ki opisujejo obravnavo koncentracije represorja (a) in spremembo koncentracije mRNA (b).



Vrednost mehke spremenljivke *mRNAIncrease* določimo na podlagi vrednosti mehke spremenljivke *Repressor* in mehkih IF-THEN pravil. Glede na lingvistični opis zakonitosti dinamike procesa transkripcije definiramo naslednji nabor mehkih IF-THEN pravil:

$$\begin{aligned}
 r_1: & \text{ IF } (\text{Repressor is } \overline{\text{ABSENT}}) \text{ THEN } (\text{mRNAIncrease is } \overline{\text{HIGH}}), \\
 r_2: & \text{ IF } (\text{Repressor is } \overline{\text{PRESENT}}) \text{ THEN } (\text{mRNAIncrease is } \overline{\text{LOW}}). \quad (6.5)
 \end{aligned}$$

Model represitorja s tremi geni, v katerem smo proces transkripcije opisali z mehkim modelom, smo zgradili v okolju MATLAB Simulink® [51]. Diagram modela prikazuje slika 6.7, na katerem je mehki model procesa transkripcije, ki je nadomestil model procesa transkripcije na osnovi diferencialne enačbe, obkrožen z rdečo. Ostali segmenti modela so nespremenjeni glede na model s slike 6.2.



Slika 6.7

Model represitorja s tremi členi, v okolju MATLAB Simulink® [51]. Z rdečo je obkrožen mehki model procesa transkripcije, ki je nadomestil model transkripcije na osnovi diferencialne enačbe, z modro proces degradacije mRNA, z zeleno proces translacije in z rumeno proces degradacije protei- nov.

Nabor vrednosti vhodne in izhodne spremenljivke bi moral obsegati vse možne vrednosti, ki ju ti dve spremenljivki v svoji ostri obliki za biološki sistem, ki ga modeliramo, lahko zavzmeta (npr. koncentracija represorja pri represitorju lahko zavzema vrednosti od 0 do približno 400 nM). Vrednosti vhodnih in izhodnih spremenljivk zaradi lažje predstavitve z mehko logiko in poudarka na relativne odvisnosti vhodne in izhodne spremenljivke definiramo na enotnih intervalih [0, 1], kjer 0 predstavlja najnižjo možno koncentracijo, 1 pa najvišjo.

Naš pristop uporabimo kot dopolnitev obstoječe metode, za katero velja, da se vrednosti vhodnih in izhodnih spremenljivk ponavadi ne nahajajo na intervalu [0, 1],

ampak so za primer represilatorja koncentracije posameznih proteinov v velikostnem redu 100 nM, medtem ko so najvišje koncentracije ustreznih mRNA ponavadi nekajkrat nižje. V ta namen vpeljemo funkciji *prescale*, ki vrednost vhodne spremenljivke v ostri obliki pred mehčanjem preslika na interval $[0, 1]$, in *postscale*, ki vrednost z intervala $[0, 1]$ preslika v končno vrednost izhodne spremenljivke v ostri obliki. S pomočjo omenjenih funkcij na podlagi izkustvenega poznavanja maksimalne koncentracije določi področni strokovnjak in vrednost vhodne ter izhodne spremenljivke umesti v biološko relevanten interval. S tem dosežemo kompatibilnost vrednosti ostrega dela modela z mehkim modelom tako na vhodu kot tudi izhodu, za predstavitev pripadnostnih funkcij pa uporabljamo relativne vrednosti na intervalih $[0, 1]$. Funkciji sta predstavljeni z algoritmoma 1 in 2. Slednja glede na maksimalno koncentracijo proteina na vhodu in spremembo koncentracije mRNA na izhodu mehkega modela vršita normalizacijo. Z novo vpeljanima funkcijama dosežemo, da bosta najvišja koncentracija represorja in najvišja možna sprememba mRNA v skladu z ostalimi parametri v sistemu. S tem dosežemo, da naš mehki model transkripcije kljub neznanim vrednostim parametrov k_{tsk} in n proces opiše kvantitativno in se lahko združi z referenčnim modelom.

6.5 Združevanje mehkega modela in modela na podlagi diferencialnih enačb

Stanje biološkega sistema v časovni točki t je v modelih obstoječih metod za kvantitativno modeliranje predstavljeno z vektorjem koncentracij N opazovanih kemijskih zvrsti $\mathbf{x}(t) = (x_1(t), x_2(t), \dots, x_N(t))$. Spremembo stanja sistema v metodah, ki temeljijo na sistemu diferencialnih enačb, določa izraz

$$\frac{d[x_i(t)]}{dt} = \sum_{j=1}^m f_{i,j}(\mathbf{x}(t)), \quad i = 1, \dots, N, \quad (6.6)$$

kjer vsaka funkcija f v vsoti opisuje enega od m bioloških procesov (npr. proces transkripcije, translacije, degradacije itd.) in ima svoj nabor kinetičnih parametrov, ki pogojujejo dinamiko procesa (npr. hitrost transkripcije, hitrost translacije, hitrost degradacije itd.). V primeru neznanih vrednosti kinetičnih parametrov opis enega od procesov nadomestimo z mehkim opisom

Algoritem 1

Prescale funkcija za preslikavo vrednosti koncentracij na interval $[0, 1]$.

$prescale_{in}$ = trenutna vrednost, ki določa maksimalno koncentracijo represorja;
 $prescale_{out}$ = nova vrednost, ki določa maksimalno koncentracijo represorja;
 $input$ = trenutna koncentracija represorja (vhod v naš mehki model);
 $input_{prescaled}$ = trenutna koncentracija represorja preslikana na interval $[0, 1]$, glede na zadnjo popravljeno vrednost, ki določa maksimalno koncentracijo represorja;

function PRESCALE($prescale_{in}, input$)

$prescale_{out} \leftarrow prescale_{in};$

if $input > prescale_{in}$ *then*
 $prescale_{out} \leftarrow input;$
end if

$input_{prescaled} \leftarrow input / prescale_{out};$
return $[input_{prescaled}, prescale_{out}];$
end function

Algoritem 2

Postscale funkcija za preslikavo vrednosti koncentracij iz intervala $[0, 1]$.

$postscale_{in}$ = trenutna vrednost, ki določa maksimalno povečanje koncentracije mRNA;
 $postscale_{out}$ = nova vrednost, ki določa maksimalno povečanje koncentracije mRNA;
 $output$ = povečanje koncentracije mRNA, predstavljeno na intervalu $[0, 1]$ (izhod iz našega mehkega modela);
 $output_{postscaled}$ = povečanje koncentracije mRNA, preslikano iz intervala $[0, 1]$ glede na $postscale_{out}$;
 $multiplier$ = ocena (velikostni razred) maksimalne koncentracije mRNA;

function POSTSCALE($postscale_{in}, output, multiplier$)

$postscale_{out} \leftarrow postscale_{in};$

if $output \cdot multiplier > postscale_{out}$ *then*
 $postscale_{out} \leftarrow output \cdot multiplier;$
end if

$output_{postscaled} \leftarrow output \cdot postscale_{out};$
 $return [output_{postscaled}, postscale_{out}];$
end function

$$\frac{d[x_i(t)]}{dt} = FL_k(\mathbf{x}(t)) + \sum_{j=1}^{m-1} f_{ij}(\mathbf{x}(t)), \quad i = 1, \dots, N, \quad k \neq j, \quad (6.7)$$

kjer je $FL_k(\mathbf{x}(t))$ prispevek spremembe koncentracije, ki ga izračunamo z mehkim modelom opazovanega biološkega procesa. Vhodne spremenljivke mehkega modela so ostro predstavljene koncentracije opazovanih kemijskih zvrsti $\mathbf{x}(t)$, izhodna spremenljivka pa ostra oblika spremembe koncentracije i -te kemijske zvrsti $d[x_i(t)]$. Izhod mehkega modela pri izračunu novega stanja sistema v vsakem diskretnem simulacijskem časovnem koraku uporabimo kot prispevek enega od procesov, ki vpliva na spremembo koncentracije, pri čemer prispevke ostalih procesov izračunamo z modelom na osnovi sistema diferencialnih enačb. Na enak način kot se velikost simulacijskega časovnega koraka dt pri numeričnem reševanju upošteva za prispevke modela na osnovi diferencialnih enačb, se ta upošteva tudi za prispevek mehkega modela.

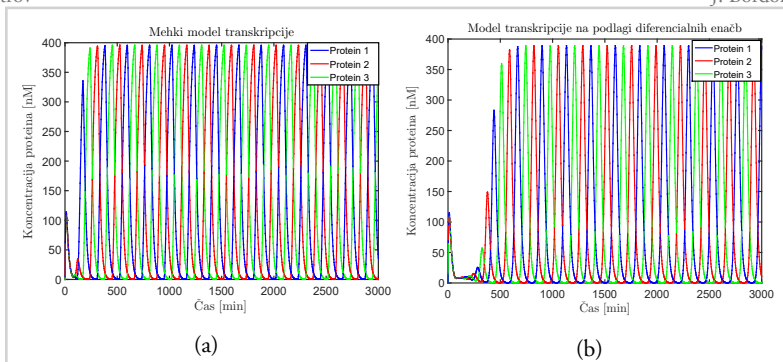
6.6 Rezultati

Tako referenčni kot tudi mešani mehko-deterministični model smo zgradili v programskem okolju MATLAB Simulink® [51]. Shemi obeh modelov sta prikazani na slikah 6.2 in 6.7. Vse simulacije smo izvajali pod istimi pogoji, in sicer smo za numerično integriranje uporabljali simulacijsko metodo *ode4* (*Runge-Kutta*) z nespremenljivim časovnim korakom 0.1 minute. Kadar ni navedeno drugače, smo uporabljali vrednosti parametrov iz tabele 6.1, ki so povzeti po [8]. Simulirali smo 3000 minut dinamike sistema in opazovali prisotnost oziroma odsotnost oscilacij koncentracij treh proteinov ter njihovo amplitudo in frekvenco. Simulacijski rezultati obeh modelov so prikazani na sliki 6.8.

Za natančnejšo primerjavo simulacijskih rezultatov in verifikacijo mehkega modela transkripcije smo na rezultatih simulacij pri različnih naborih vrednosti sicer znanih parametrov izvedli frekvenčno analizo s Hitro Fourierjevo transformacijo (angl. *Fast Fourier transform* - FFT). S slednjo smo opazovali, kako uporaba mehkega modela procesa transkripcije vpliva na frekvenco in amplitudo oscilacij. Glede na to, da popolne natančnosti modelov na podlagi sistema diferencialnih enačb ne moremo doseči brez natančnih vrednosti vseh kinetičnih parametrov, ki pogojujejo dinamiko, smo pričakovali, da bomo z uporabo mehkega modela v sistem vpeljali manjšo napako. Za vsak

Slika 6.8

Simulacijski rezultati re-presilatorja s tremi členi modela, pri katerem smo transkripcijo opisali z mehko logiko (a) in modela, pri katerem smo uporabili izključno sistem diferencialnih enačb (b). Pri obeh modelih koncentracije proteinov stabilno oscilirajo s primerljivo frekvenco in amplitudo.

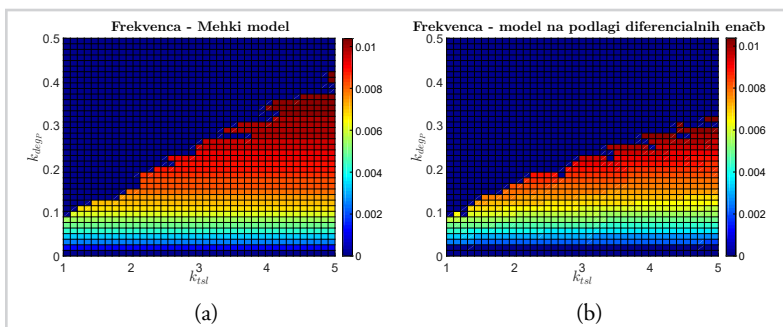


proces, ki ga namesto z obstoječimi metodami opišemo z uporabo našega pristopa, pričakujemo, da bo natančnost modela manjša, a kljub temu kvantitativno relevantna.

Rezultati frekvenčne analize, predstavljeni na sliki 6.9, prikazujejo simulacije pri različnih vrednostih parametrov k_{degp} in k_{tsl} , pri katerih koncentracije oscilirajo. Nabor vrednosti, pri katerih koncentracije oscilirajo, je pri mehkem modelu procesa transkripcije nekoliko širši. Kljub temu sta frekvenca in amplituda oscilacij povsem primerljivi s simulacijskimi rezultati referenčnega modela.

Slika 6.9

Rezultati frekvenčne analize za parametra k_{tsl} in k_{degp} na intervalih $[1, 5 \text{ min}^{-1}]$ in $[0, 0.5 \text{ min}^{-1}]$.



Rezultati simuliranja referenčnega modela pri različnih vrednostih hitrosti transkripcije k_{tsk} pokažejo, da se velikost področja pri katerih se oscilacije pojavljajo, frekvence slednjih pa ohranjajo, ne spreminja. Področje oscilacij torej ostaja primerljivo s področjem, ki ga pridobimo z rezultati simuliranja procesa transkripcije z mehkim modelom.

S spreminjanjem vrednosti hitrosti transkripcije se linearno povečuje velikost amplitude, česar z mehkim modelom procesa transkripcije ni mogoče zajeti. Kljub temu so za biološko relevanten nabor vrednosti hitrosti transkripcije amplitude oscilacij primerljive. Razlika med referenčnim modelom in mehkim modelom je glede na različne nabore vseh parametrov največja predvsem v področjih bifurkacij, tj. področjih, v katerih sistem prehaja iz oscilatornega v stabilno stanje. Frekvenčna analiza torej potrdi, da uporaba mehkega modela procesa transkripcije ne spremeni kvantitativne relevantnosti simulacijskih rezultatov glede na vrednosti preostalih kinetičnih parametrov kljub predpostavki, da hitrosti transkripcije k_{tsk} in Hillovega koeficienta n pri mehkiem modelu ne poznamo.

6.7 Vrednotenje metode in diskusija

Izgradnja mehkega modela temelji na izkustvenem poznavanju sistema, ki ga želimo modelirati. Izkustveno poznavanje sistema lahko z uporabo mehke logike pri izgradnji mehkega modela vključimo na način, ki pri ostalih obstoječih metodah ni mogoč. Z mehkim modelom se izognemo uporabi natančnih vrednosti kinetičnih parametrov, ki določajo dinamične zakonitosti procesov, a po drugi strani vpeljemo nove parametre. Dva takšna parametra sta maksimalna koncentracija in sprememba koncentracije, ki ju pri opisu transkripcije uporabljamo v funkcijah *prescale* (npr. interval $[0, 400 \text{ nM}]$) in *postscale* (npr. interval $[0, 1.2 \text{ nM}]$), ostali parametri pa nastopajo v definicijah pripadnostnih funkcij vhodne in izhodne spremenljivke mehkega modela procesa transkripcije. Kljub temu, da z uporabo mehke logike vpeljemo večje število parametrov, pa vrednosti le-teh na podlagi ekspertnega znanja določimo bistveno lažje kot natančne vrednosti kinetičnih parametrov, ki jih potrebujemo za pridobitev simulacijskih rezultatov z uporabo obstoječih metod. Takšna parametrizacija modela je torej bližje končnemu uporabniku. Vzpostavljena metoda ni namenjena izgradnji mehkega modela, pri katerem bi z iskanjem optimalnih vrednosti njegovih parametrov dosegli popolno ujemanje z referenčnim modelom. Z metodo, ki temelji na mehki logiki, lahko pri izgradnji modela uporabimo ekspertno znanje področnih strokovnjakov, kar nam omogoča pridobitev biološko relevantnih rezultatov tudi v primeru, ko natančnih vrednosti kinetičnih parametrov ne poznamo ali jih ne znamo oceniti. Rezultati mehkega modela se torej v primeru zadostnega izkustvenega poznavanja sistema do neke mere ujemajo z rezultati referenčnega modela, medtem ko pričakujemo, da se rezultati mehkega modela, zgrajenega na podlagi napačnega izkustvenega poznavanja, izrazito

razlikujejo od rezultatov referenčnega.

6.7.1 Modeli in metrike za vrednotenje natančnosti mehkega modela

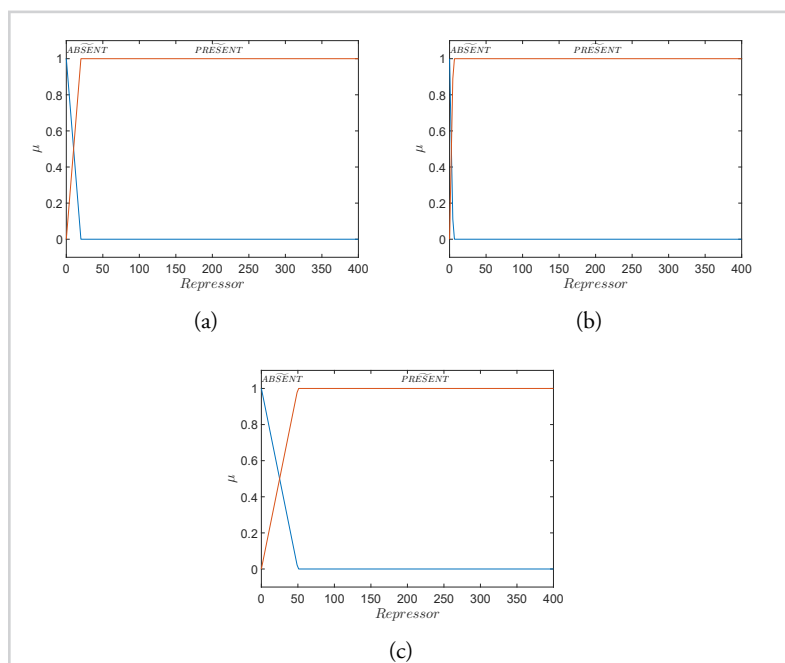
Za potrebe vrednotenja mehkega modela smo definirali tri različice mehkega modela transkripcije, pri katerih smo uporabili tri različna izkustvena poznavanja procesa transkripcije. Mehki model iz razdelka 6.4 poimenujemo $FL_{osnovni}$, pri mehkem modelu $FL_{močnejši}$ izkustveno poznavanje predpostavlja močnejšo, pri mehkem modelu $FL_{šibkejši}$ pa šibkejšo vezavo represorja v primerjavi z osnovnim modelom. Moč vezave v mehkem modelu opišemo z izbiro oblik pripadnostnih funkcij vhodne spremenljivke *Repressor*. Pripadnostne funkcije treh različic mehkega modela prikazuje slika 6.10. Kljub temu, da smo v razdelku 6.4.4 vhodne in izhodne vrednosti v mehki model zaradi poenostavitve definirali na intervalu $[0, 1]$, smo za vrednotenje pri vseh treh modelih predpostavili, da se vrednost vhodne spremenljivke nahaja na intervalu $[0, 400 \text{ nM}]$, vrednost izhodne spremenljivke pa na intervalu $[0, 1.2 \text{ nM}]$, medtem ko smo pripadnostne funkcije izhodne spremenljivke *mRNAIncrease* in mehka IF-THEN pravila povzeli po mehkem modelu iz razdelka 6.4.

Definirali smo tudi tri različice modela transkripcije na podlagi funkcije iz izraza (6.3), ki se razlikujejo v vrednosti Hillovega koeficienta n ($ODE_{n=0.5}$, $ODE_{n=2}$ in $ODE_{n=4}$), s katerimi si pomagamo pri vrednotenju razlike med mehkimimi modeli in referenčnim modelom ($ODE_{n=2}$). Napako posameznih mehkih modelov ter modelov $ODE_{n=0.5}$ in $ODE_{n=4}$ določimo na podlagi integralne kvadratne napake, ki jo za diskretni signal izračunamo na podlagi izraza

$$ISE = \sum_{i=0}^{N-1} (\hat{x}_i - x_i)^2, \quad (6.8)$$

kjer $(\hat{x}_i - x_i)^2$ predstavlja kvadratno napako med referenčnim (\hat{x}_i) in vrednotenim (x_i) signalom v diskretnem časovnem koraku i , N pa predstavlja število vzorcev signalov. Natančnost modela vrednotimo na podlagi:

- razlike v odzivih modelov pri vrednostih vhodne spremenljivke $[P]$ iz intervala $[0, 20 \text{ nM}]$,
- razlike v časovnih odzivih modelov, v primeru, da ima vrednost vhodne spremenljivke $[P]$ obliko invertiranega pulza na intervalu $[0, 400 \text{ nM}]$,



Slika 6.10

Pripadnostne funkcije
vходne spremenljivke
Repressor mehkih mode-
lov $FL_{povzemi}$ (a), $FL_{močnejši}$
(b) in $FL_{žibkejši}$ (c).

- razlike v simulacijskih rezultatih referenčnega modela v primerjavi s simulacijskimi rezultati preostalih modelov.

Z vpeljavo mehke logike zaradi drugačnega opisa dinamike transkripcije v model vnesemo določeno napako. Pričakujemo, da bo razlika v odzivih mehkih modelov transkripcije in modelov transkripcije na osnovi diferencialnih enačb relativno velika, razlika v simulacijskih rezultatih celotnega modela pa manjša. Slednja je namreč odvisna od dinamike delovanja vseh procesov, ki v modelu nastopajo. S spremembo enega od procesov torej spremenimo tudi dinamiko ostalih in je zato poleg odziva posameznega procesa smiselno opazovati tudi spremembo simulacijskih rezultatov celotnega modela. Pri vrednotenju bomo opazovali, ali z različnimi izkustvenimi poznavanji o moči vezave represorja v model vnesemo napako, ki ustreza biološkemu pomenu različnih moči vezav represorja. Oženje (oziroma širjenje) pripadnostnih funkcij ima na model transkripcije enak vpliv kot večanje (oziroma manjšanje) vrednosti parametra n . Ne

zanima nas torej absolutno ujemanje odzivov, ampak vpliv različnih tipov modelov na simulacijske rezultate celotnega modela. Kot metrike za vrednotenje slednjih smo poleg integralne kvadratne napake izbrali tudi amplitudo, frekvenco in moč oscilacij. Slednjo izračunamo na podlagi izraza za moč diskretnega periodičnega signala povzete tega po [75]

$$P_x = \frac{1}{N} \sum_{i=0}^{N-1} |x_i|^2,$$

(6.9)

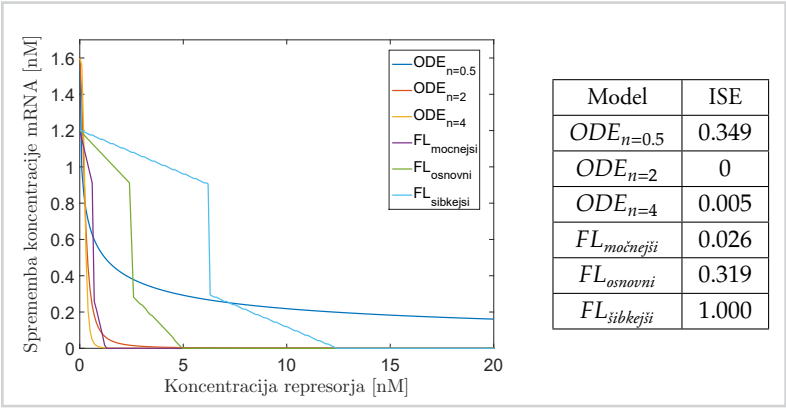
kjer N predstavlja število vzorcev v eni periodi oscilacije in x_i vrednost signala v diskretnem koraku i .

6.7.2 Odzivi modelov pri različnih vrednostih vhodne spremenljivke

V pričujočem razdelku izračunamo odziv modelov transkripcije pri vrednostih vhodne spremenljivke na intervalu $[0, 20 \text{ nM}]$, na katerem sprememba koncentracije mRNA ni enaka nič. Graf na levi strani slike 6.11 prikazuje odzive vseh šestih modelov, medtem ko tabela na desni strani iste slike prikazuje integralne kvadratne napake petih modelov v primerjavi z referenčnim modelom ($ODE_{n=2}$). Integralne napake smo normalizirali na najvišjo vrednost in s tem dobili relativno napako glede na referenčni model.

Slika 6.11

Odzivi na vhodne vrednosti spremenljivke $[P]$ iz intervala $[0, 20 \text{ nM}]$ (levo) in tabela integralnih kvadratnih napak posameznih modelov (desno).



Iz tabele je razvidno, da smo največjo napako pri odzivu modelov na različne vrednosti

vhodne spremenljivke naredili z mehkim modelom $FL_{\text{šibkejši}}$, ki predpostavlja šibkejši represor kot v osnovnem mehkem modelu, najmanjšo pa s spremembo Hillovega koeficienta na $n = 4$. Razvidno je tudi, da z upoštevanjem drugačnega izkustvenega poznavanja v mehke modele vpeljemo različno velike napake, kar nam pove, da so rezultati mehkih modelov odvisni od izkustvenega poznavanja, ki ga pri izgradnji posameznega modela upoštevamo. Glede na odzive mehkih modelov na različne vrednosti vhodne spremenljivke ima najmanjšo integralno napako mehki model, ki predpostavlja močnejšo vezavo represorja.

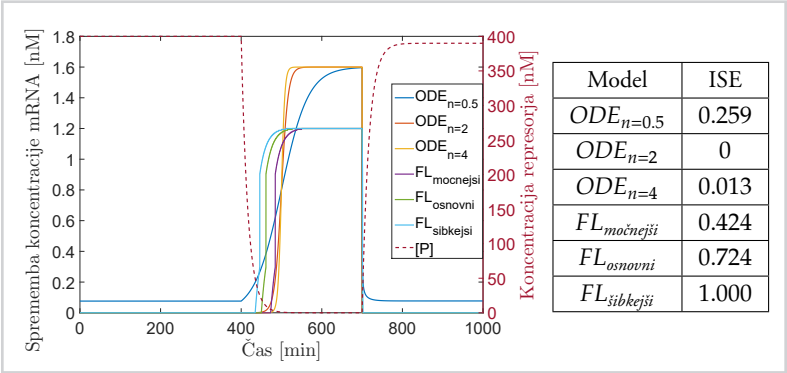
6.7.3 Časovni odzivi modelov pri invertiranem pulzu vrednosti vhodne spremenljivke

Časovni odziv modelov smo določili z invertiranim pulzom vrednosti vhodne spremenljivke na intervalu $[0, 400 \text{ nM}]$, ki predstavlja koncentracijo represorja. Prvih 400 minut je koncentracija represorja 400 nM . Po preteku 400 minut omogočimo njegovo degradacijo, ki je določena na podlagi negativnega dela izraza 6.1. Ko koncentracija represorja zaradi degradacije pade na 0 nM , odzivi vseh modelov dosežejo maksimalno vrednost spremembe koncentracije mRNA. Nato omogočimo translacijo represorja, ki njegovo koncentracijo poviša do ravnovesne lege blizu njegove maksimalne izhodiščne koncentracije in simuliramo do 1000 minut. Tudi v primeru časovnega odziva modelov smo integralne kvadratne napake zaradi lažje interpretacije normalizirali na največjo napako. Graf časovnega odziva in tabela z integralnimi kvadratnimi napakami sta prikazana na sliki 6.12. Vhodni signal je na sliki označen s črtkano črto, odzivi posameznih modelov pa s polnimi črtami različnih barv.

Podobno kot pri odzivu na različne vrednosti vhodne spremenljivke ima tudi pri časovnem odzivu na invertiran pulz vhodne spremenljivke model $FL_{\text{šibkejši}}$ v primerjavi z referenčnim največjo integralno napako, medtem ko ima najmanjšo napako model $ODE_{n=4}$. Velik del napake pri mehkih modelih lahko pripišemo časovnemu intervalu, ko je koncentracija represorja zelo blizu nič (simulacijski čas med 500 in 700 minut). V tem intervalu je, zaradi nenatančne ocene maksimalne spremembe koncentracije mRNA, pri mehkih modelih odstopanje od ODE modela visoko. To napako bi lahko v primeru natančnejšega izkustvenega poznavanja maksimalne spremembe koncentracije mRNA zmanjšali.

Slika 6.12

Časovni odzivi na in-
vertiran pulz vhodne
spremenljivke v intervalu
[0, 400 nM], ki predstavlja
koncentracijo represorja
(levo), in tabela integralnih
kvadratnih napak posa-
meznih modelov (desno).
Pri časovnem odzivu je
koncentracija represorja
označena s črtkano črto,
pri čemer njeno vrednost
razberemo iz desne ordi-
natne osi, medtem ko so
spremembe koncentracij
mRNA označene s polnimi
črtami, vrednosti pa raz-
beremo iz leve ordinatne
osi.



6.7.4 Primerjava simulacijskih rezultatov

Preverili smo tudi kako spremembe v parametru n in upoštevanje različnih tipov izku-
stvenega poznavanja spremenijo simulacijske rezultate celotnega modela represilatorja.
Simulirali smo z enakimi nastavitvami in vrednostmi kinetičnih parametrov kot v raz-
delku 6.6, pri čemer smo čas simulacije zmanjšali na 1500 minut. Rezultati vseh šestih
modelov so prikazani na sliki 6.13. Integralne napake, frekvence, amplitude in moči
oscilacij posameznih modelov so podane v tabeli 6.2. Integralne kvadratne napake so,
podobno kot v predhodnih razdelkih, normalizirane na največjo napako.

Slika 6.13

Simulacijski rezultati šestih
modelov. Iz grafa je razvi-
dno, da model $ODE_{n=0.5}$
ne izraža oscilacij, medtem
ko simulacijski rezultati
modela $ODE_{n=4}$ izrazito
odstopajo od rezultatov
preostalih modelov. Raz-
like med posameznimi
mekhimi modeli so pred-
vsem v frekvenci oscilacij.

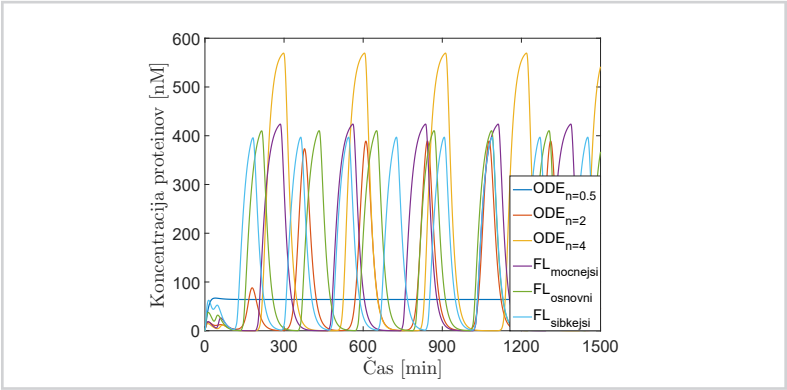


Tabela 6.2

Integralne kvadratne napake, razlike v frekvenci, amplitudi in moči oscilacij petih modelov (model $ODE_{n=0.5}$ ne izraža oscilacij).

Model	ISE	Frek./Razlika		Amp./Razlika		Moč/Razlika	
$ODE_{n=0.5}$	-	0	-	-	-	-	-
$ODE_{n=2}$	0	0.257	0 %	389	0 %	24670	0 %
$ODE_{n=4}$	1.000	0.195	-23.9 %	569	+46 %	72646	+194.5 %
$FL_{močnejši}$	0.529	0.218	-15.2 %	423	+9 %	42932	+74.0 %
$FL_{osnovni}$	0.269	0.275	+7.2 %	410	+5.5 %	41560	+68.5 %
$FL_{šibkejši}$	0.558	0.330	+28.8 %	395	+1.8 %	41329	+67.5 %

Napake pri rezultatih simulacij celotnega represilatorja se nekoliko razlikujejo od napak, ki smo jih opazili pri odzivih modelov, ki transkripcijo obravnavajo izolirano. Iz grafa je razvidno, da koncentracije proteina pri modelu $ODE_{n=0.5}$ ne oscilirajo. Integralna napaka je največja pri modelu $ODE_{n=4}$, najmanjša pa pri mehkem modelu $FL_{osnovni}$. Spremembe v amplitudi in posledično moči oscilacij so pri vseh mehkih modelih manjše kot pri modelu $ODE_{n=4}$. Amplitude oscilacij pri mehkih modelih se od referenčnega razlikujejo za manj kot 10 %, medtem ko je razlika pri frekvencah bolj izrazita. Sklepamo torej lahko, da z napačnim izkustvenim poznavanjem v primeru represilatorja vplivamo predvsem na napako v frekvenci.

6.7.5 Diskusija

Na podlagi vrednotenja vseh šestih modelov z različnimi pristopi vidimo, da po pričakovanjih napake v odzivih modelov, ki proces transkripcije obravnavajo izolirano, niso neposredno povezane z napakami, ki jih dobimo s simuliranjem celotnega modela. Dinamika sistema je odvisna od dinamike vseh procesov skupaj in ne izključno od transkripcije, zato na simulacijske rezultate celotnega modela očitno napaka samega odziva vpliva precej manj, kot sama kvalitativna oblika odziva. Pri modelih $ODE_{n=2}$ in $ODE_{n=4}$ sta namreč tako odziv na različne vrednosti vhodne spremenljivke kot tudi časovni odziv na invertiran pulz vhodne spremenljivke glede na integralno napako zelo podobna, bistveno pa se razlikujejo simulacijski rezultati obeh modelov celotnega represilatorja.

Odziv modela $ODE_{n=4}$, ki predpostavlja močnejšo vezavo represorja kot model $ODE_{n=2}$, je precej bolj strm od odzivov vseh preostalih modelov, kar pomeni, da se vre-

dnosti izhodne spremenljivke v zelo ozkem intervalu vrednosti vhodnih spremenljivk spreminjajo bistveno hitreje kot pri modelih z manj strmim odzivom. Podoben sklep lahko glede na rezultate v tabeli 6.2 naredimo tudi pri primerjavi mehkih modelov $FL_{močnejši}$ (bolj strm odziv) in $FL_{osnovni}$ (manj strm odziv). Z ožanjem pripadnostnih funkcij vhodne spremenljivke povečujemo strmino odziva mehkega modela, s čimer dosežemo enak vpliv na rezultate (zmanjšanje frekvence, povečanje amplitude in moči) kot z večanjem parametra n . Tako pri mehkem modelu kot tudi pri modelu na podlagi diferencialnih enačb spremembi ustrežata biološkemu pomenu večanja moči vezave represorja. V primeru, da na podlagi izkustvenega poznavanja napačno predpostavimo šibkejši represor (pripadnostne funkcije širimo), v model vnesemo napako, ki je skladna z biološkim pomenom nižanja moči vezave represorja (povečanje frekvence, zmanjšanje amplitude in moči glede na osnovni mehki modeli $FL_{osnovni}$). Z vpeljavo mehke logike smo v model vnesli določeno napako, o čemer pričajo integralna napaka ter spremembe frekvence, amplitude in moči glede na referenčni model $ODE_{n=2}$. Kljub temu smo z mehkim modelom transkripcije dinamiko represilatorja glede na referenčni model $ODE_{n=2}$ opisali bolj natančno kot v primeru spremembe parametra n pri modelu $ODE_{n=0.5}$ in $ODE_{n=4}$.

Razlike med simulacijskimi rezultati mehkih modelov pripišemo različnim postopkom izgradnje. Mehki model $FL_{osnovni}$ smo zgradili na podlagi poglobljene analize transkripcije referenčnega modela (glej razdelek 6.4.2), medtem ko smo pri preostalih dveh mehkih modelih predpostavili povsem drugačna tipa represorja, ki ne ustrežata analizi. Napaka pri simulacijskih rezultatih modelov $FL_{močnejši}$ in $FL_{šibkejši}$ je glede na referenčni model $ODE_{n=2}$ po pričakovanjih večja kot pri simulacijskih rezultatih mehkega modela $FL_{osnovni}$.

Z vrednotenjem šestih modelov smo pokazali, da na podlagi izkustvenega poznavanja lahko zgradimo mehki model, s katerim pridobimo biološko relevantne simulacijske rezultate. Po drugi strani smo pokazali, da v primeru uporabe napačnega izkustvenega poznavanja v mehki model vnesemo napako, ki se skladno z modeli na podlagi diferencialnih enačb izrazi kot napaka pri simulacijskih rezultatih. Takšna občutljivost mehkega modela na spremembe nam torej potencialno omogoča tudi natančnejše nastavljanje lastnosti mehke logike (pripadnostnih funkcij) glede na eksperimentalne podatke, kadar so ti na voljo.

Vpeljava mehke logike v zvezne Petrijeve mreže

7

V pričujočem poglavju predstavimo zvezne Petrijeve mreže z mehko logiko, ki združujejo zvezne Petrijeve mreže z metodo izgradnje mehkih modelov bioloških procesov, predstavljeno v poglavju 6. Zvezni Petrijevi mreži za potrebe modeliranja z mehko logiko dodamo mehke prožilne funkcije, ki jih uporabimo za mehki opis procesov gensko regulatornega omrežja, v katerem nekatere vrednosti kinetičnih parametrov niso znane. V prvem delu poglavja predstavimo osnove Petrijevih mrež, nadaljujemo pa s pregledom področja modeliranja bioloških sistemov s Petrijevimi mrežami. Nato definiramo zvezne Petrijeve mreže z mehko logiko (angl. *Continuous Petri nets with fuzzy logic* - CPNFL) ter mehke prožilne funkcije za posamezne procese, ki predstavljajo izvirni znanstveni doprinos pričujoče disertacije. Njihovo delovanje in uporabo predstavimo na primeru izgradnje modela cirkadianega ritma modelnega organizma glive *Neurospora* [76].

7.1 Petrijeve mreže

Osnovno obliko Petrijevih mrež je leta 1939 prvič uporabil, leta 1962 pa definiral nemški matematik Carl Adam Petri. Razvile so se v matematični modelirni jezik, ki se uporablja kot univerzalno orodje za modeliranje dinamičnih distribuiranih sistemov na različnih področjih. Za podrobno razlago lastnosti Petrijevih mrež ter predstavitev njihove uporabe na različnih primerih priporočamo vir [77]. Petrijeve mreže se uporabljajo tako za deterministično kot tudi stohastično modeliranje distribuiranih in paralelnih sistemov ter procesov. Uveljavile so se kot orodje za predstavitev in modeliranje končnih avtomatov, paralelnih aktivnosti v proizvodni industriji, podatkovne pretočnosti, komunikacijskih protokolov, nadzora sinhronizacije, formalnih jezikov, analize delovanja večprocesorskih sistemov itd.

7.1.1 Osnovna definicija Petrijevih mrež

Osnovna definicija Petrijevih mrež je povzeta po viru [78].

Definicija 3: Petrijeva mreža je četverček $PN = (P, T, I, O)$:

- $P = \{p_1, p_2, \dots, p_n\}$ je končna množica vseh pogojev (angl. *places*), $n \geq 0$.
- $T = \{t_1, t_2, \dots, t_k\}$ je končna množica vseh akcij (angl. *transitions*), $k \geq 0$.

- Velja $P \cap T = \emptyset$.
- $I : T \rightarrow P^\infty$ in $P \rightarrow T^\infty$ sta funkciji vhoda, ki določata večkratno množico vhodnih pogojev za vse akcije v T in večkratno množico vhodnih akcij za vse pogoje v P .
- $O : T \rightarrow P^\infty$ in $P \rightarrow T^\infty$ sta funkciji izhoda, ki določata večkratno množico izhodnih pogojev za vse akcije v T in večkratno množico izhodnih akcij za vse pogoje v P .

Pogoj p_i je vhodni, če velja $p_i \in I(t_j)$ oziroma izhodni, če velja $p_i \in O(t_j)$. Pogoj se lahko v večkratnih množicah $I(t_j)$ in $O(t_j)$ pojavi večkrat. Število pojavitev za vhodne pogoje zapišemo z izrazom

$$\#(p_i, I(t_j)), \quad (7.1)$$

za izhodne pogoje pa z izrazom

$$\#(p_i, O(t_j)). \quad (7.2)$$

Za akcijo t_j velja, da je vhodna (izhodna) akcija pogoja p_i , če velja, da je p_i izhodni (vhodni) pogoj akcije t_j . Število pojavitev vhodnih in izhodnih akcij v večkratnih množicah $I(p_i)$ in $O(p_i)$ zapišemo z izrazom

$$\#(t_j, I(p_i)) \quad (7.3)$$

za vhodne akcije in

$$\#(t_j, O(p_i)) \quad (7.4)$$

za izhodne akcije. Za predstavitev večkratnih množic vhodnih in izhodnih akcij (pogojev) uporabljamo sledeči krajši zapis:

- $\bullet t_j = I(t_j)$ je večkratna množica vhodnih pogojev akcije t_j ,

- $t_j^\bullet = O(t_j)$ je večkratna množica izhodnih pogojev akcije t_j ,
- $\bullet p_i = I(p_i)$ je večkratna množica vhodnih akcij pogoja p_i ,
- $p_i^\bullet = O(p_i)$ je večkratna množica izhodnih akcij pogoja p_i .

Primer Petrijeve mreže s tremi pogoji in eno akcijo podamo z naslednjimi naborem izrazov

$$PN = (P, T, I, O), \quad (7.5)$$

$$P = \{p_1, p_2, p_3\}, \quad (7.6)$$

$$T = \{t_1\}, \quad (7.7)$$

$$\bullet t_1 = \{p_1, p_2\}, \quad (7.8)$$

$$t_1^\bullet = \{p_3\}. \quad (7.9)$$

7.1.2 Graf Petrijeve mreže

Grafično predstavitev Petrijeve mreže podamo z grafom Petrijeve mreže. Slednji je predstavljen kot dvodelni usmerjeni multigraf, pri katerem točke razdelimo v dve nepresečni množici, t.j. pogoje in akcije. Pogoje v grafu Petrijeve mreže predstavimo s krogi, akcije pa z ravno črto ali pravokotnikom. Povezave v grafu Petrijeve mreže povezujejo bodisi pogoj z akcijo bodisi akcijo s pogojem. Graf ima lahko med pogojem p_i in akcijo t_j več vzporednih enako usmerjenih povezav, njihovo število pa je določeno s številom pojavitev

- $\#(p_i, \bullet t_j)$ pri povezavah usmerjenih od pogoja do akcije ali
- $\#(p_i, t_j^\bullet)$ pri povezavah usmerjenih od akcije do pogoja.

Definicija 4: Graf Petrijeve mreže $G = (V, A)$ je dvodelni usmerjeni multigraf, kjer sta $V = \{v_1, v_2, \dots, v_s\}$ množica vozlišč in $A = \{a_1, a_2, \dots, a_r\}$ večkratna množica usmerjenih povezav $a_i = (v_k, v_l)$, za katere velja $v_k, v_l \in V$. Množico V lahko razdelimo v dve tuji si množici P in T , tako da velja $V = P \cup T$ in $P \cap T = \emptyset$. Za

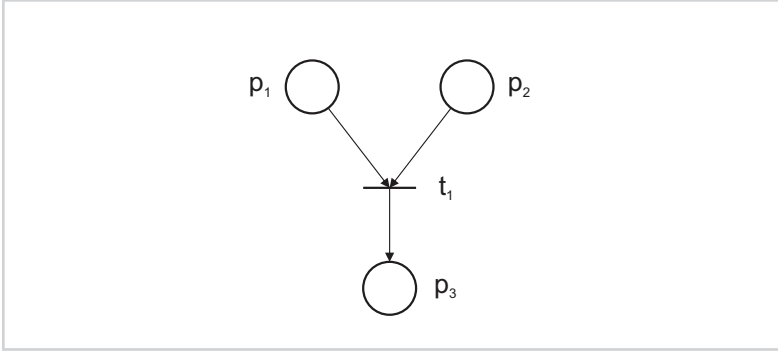
t_1 t_1

Modeliranje bioloških sistemov z uporabo mehke logike in Petrijevih mrež

 p_3 p_3 δ_I

vsako usmerjeno povezavo $a_i = (v_k, v_l)$ velja bodisi $v_k \in P$ in $v_l \in T$ bodisi $v_l \in P$ in $v_k \in T$.

Primer grafa Petrijeve mreže, ki ustreza Petrijevi mreži definirani z naborom izrazov (7.5)-(7.9), prikazuje slika 7.1.



Slika 7.1

Graf Petrijeve mreže, ki ustreza Petrijevi mreži definirani z naborom izrazov (7.5)-(7.9).

7.1.3 Označitev Petrijeve mreže

Označitev (angl. *marking*) m predstavlja dodelitev žetonov (angl. *token*) v pogoje Petrijeve mreže. Označitev predstavimo z vektorjem \mathbf{m} , ki ga zapišemo z izrazom

$$\mathbf{m} = (m(p_1), m(p_2), \dots, m(p_{|P|})), \quad (7.10)$$

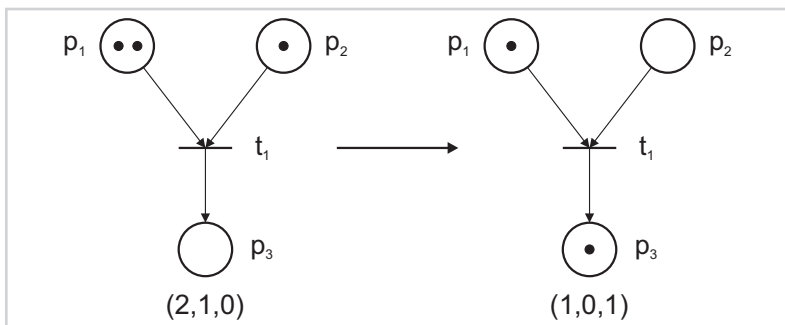
kjer je $m(p_i) \in \mathbb{N}_0$ označitev pogoja p_i , $i = 1, 2, \dots, |P|$, $|P|$ pa predstavlja moč množice P . Izvajanje (angl. *execution*) Petrijeve mreže lahko označitev spremeni, pri čemer se ob proženju akcije lahko število žetonov v pogojih poveča, zmanjša ali ostane enako. Izvajanje Petrijeve mreže je odvisno od strukture Petrijeve mreže in števila ter razporeditve žetonov v Petrijevi mreži, vrši pa se s proženjem akcij. Slednje povzroči prenos žetonov iz vhodnih pogojev v izhodne pogoje prožene akcije. Akcija se lahko proži, če je omogočena. Akcija t_j je omogočena, če za vsak pogoj $p_i \in P$ velja

$$m(p_i) \geq \#(p_i, \bullet t_j). \quad (7.11)$$

Akcija, ki je omogočena, se lahko (ne pa nujno) proži. Ob proženju akcije t_j se novo označitev za vsak pogoj $p_i \in P$ izračuna na podlagi izraza

$$m(p_i) = m(p_i) - \#(p_i, \bullet t_j) + \#(p_i, t_j^\bullet). \quad (7.12)$$

V grafu Petrijeve mreže žetone predstavimo s črnimi pikami v pogojih. Proženje akcije t_1 Petrijeve mreže, ki smo jo definirali z naborom izrazov (7.5)-(7.9) in ima začetno označitev $\mathbf{m} = (2, 1, 0)$, prikazuje graf Petrijeve mreže na sliki 7.2.



Slika 7.2

Graf Petrijeve mreže, ki predstavlja Petrijevo mrežo z eno akcijo in tremi pogoji pred (levo) in po (desno) proženju akcije t_1 .

Proženje akcije v Petrijevih mrežah je hipno. Odvisno od načina izvajanja Petrijeve mreže se lahko naenkrat proži ena ali več omogočenih akcij. V primeru proženja ene akcije je izbira slednje odvisna od tipa izvajanja (izbira na podlagi naključne spremenljivke, izbira na podlagi prioritete vrste itd.).

7.2 Mehke Petrijeve mreže

Definicije mehkih Petrijevih mrež (angl. *Fuzzy Petri Net* - FPN) so bile vzpostavljene za potrebe uporabe metodologije mehke logike v Petrijevih mrežah [79–83]. Definicijo mehkih Petrijevih mrež v pričujoči disertaciji povzemamo po viru [80].

Definicija 5: Mehka Petrijeva mreža je definirana kot $FPN = (P, T, D, I, O, f, \alpha, \beta)$:

- $P = \{p_1, p_2, \dots, p_n\}$ je končna množica vseh pogojev, $n \geq 0$,
- $T = \{t_1, t_2, \dots, t_k\}$ je končna množica vseh akcij, $k \geq 0$,

- Velja $P \cap T = \emptyset$,
- $D = \{d_1, d_2, \dots, d_n\}$ je končna množica vseh lingvističnih izrazov, $P \cap T \cap D = \emptyset$, $|P| = |D|$,
- $I : T \rightarrow P^\infty$ je funkcija vhoda, ki določa večkratno množico vhodnih pogojev za vse akcije v T ,
- $O : T \rightarrow P^\infty$ je funkcija izhoda, ki določa večkratno množico izhodnih pogojev za vse akcije v T ,
- $f : T \rightarrow [0, 1]$ je funkcija, ki akcijam v T dodeli vrednost med 0 in 1,
- $\alpha : P \rightarrow [0, 1]$ je funkcija, ki pogojem v P dodeli vrednost med 0 in 1,
- $\beta : P \rightarrow D$ je funkcija, ki pogojem v P dodeli lingvistični izraz iz D .

Za večkratni množici vhodnih in izhodnih pogojev akcije t_j uporabljamo enako notacijo ($\bullet t_j$ in t_j^\bullet) kot pri osnovnih Petrijevih mrežah, definiranih v razdelku 7.1.1. Za grafično ponazoritev mehke Petrijeve mreže uporabljamo definicijo grafa Petrijeve mreže iz razdelka 7.1.2. Akciji t_j v mehki Petrijevi mreži je dodeljena vrednost $f(t_j) = \mu_j$, $\mu_j \in [0, 1]$, pogoju p_i pa lingvistični izraz $\beta(p_i) = d_i$, $d_i \in D$.

Označitev pogoja p_i v mehki Petrijevi mreži je določena z $\alpha(p_i)$, pri čemer velja $\alpha(p_i) \in [0, 1]$. V primeru, da velja $\alpha(p_i) = y_i$, $y_i \in [0, 1]$ in $\beta(p_i) = d_i$, pravimo, da je resničnost lingvističnega izraza d_i enaka y_i . Primer uporabe mehke Petrijeve mreže prikažemo na pravilu r_{12} iz poglavja 4, ki ga zapišemo z izrazom

$$r_{12} : \text{ IF } (Temp \text{ is } \widetilde{\text{COMFORTABLE}}) \text{ AND } (Delta_temp \text{ is } \widetilde{\text{NEGATIVE SMALL}}) \\ \text{ THEN } (Heat_power \text{ is } \widetilde{\text{MEDIUM}}) \quad (CF = 1)$$

Mehko Petrijevo mrežo za pravilo r_{12} definiramo z naslednjim naborom izrazov

$$\begin{aligned}
PN &= (P, T, D, I, O, f, \alpha, \beta) \\
P &= \{p_1, p_2, p_3\} \\
T &= \{t_1\} \\
D &= \{Temp \text{ is COMFORTABLE}, Delta_temp \text{ is NEGATIVE SMALL}, \\
&\quad Heat_power \text{ is MEDIUM}\} \\
\bullet t_1 &= \{p_1, p_2\}, t_1^\bullet = \{p_3\}, f(t_1) = 1 \\
\alpha(p_1) &= 0'6, \alpha(p_2) = 0'57 \\
\beta(p_1) &= Temp \text{ is COMFORTABLE} \\
\beta(p_2) &= Delta_temp \text{ is NEGATIVE SMALL} \\
\beta(p_3) &= Heat_power \text{ is MEDIUM}
\end{aligned} \tag{7.13}$$

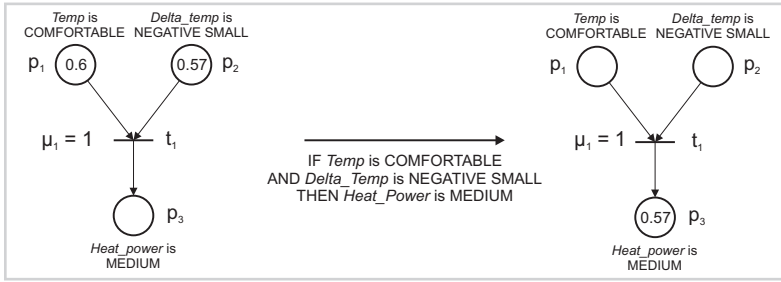
Podobno kot pri izvajanju osnovnih Petrijevih mrež se akcija lahko proži le v primeru, če je omogočena. Akcija t_j je omogočena, če velja $\forall p_i \in \bullet t_j, \alpha(p_i) \geq \lambda$, kjer je λ mejna vrednost, ki jo glede na opisno izkustveno poznavanje problemske domene določimo sami in za katero velja $\lambda \in [0, 1]$. V primeru, da ima akcija več vhodnih pogojev, ki se v pravilu povezujejo z veznikom AND, se resničnost lingvističnega izraza $\alpha(p_k) = y_k$ v izhodnem pogoju p_k izračuna na podlagi izraza

$$y_k = \min(y_{j1}, y_{j2}, \dots, y_{jn}) * \mu_j, \tag{7.14}$$

v primeru povezovanja z veznikom OR pa na podlagi izraza

$$y_k = \max(y_{j1}, y_{j2}, \dots, y_{jn}) * \mu_j, \tag{7.15}$$

kjer so $y_{j1}, y_{j2}, \dots, y_{jn}$ stopnja veljavnosti (angl. *degree of truth*) lingvističnih izrazov, ki pripadajo vhodnim pogojem akcije t_j , pri čemer velja $f(t_j) = \mu_j$. Proženje akcije mehke Petrijeve mreže, ki je definirana z naborom izrazov (7.13), prikazuje graf Petrijeve mreže na sliki 7.3.



Slika 7.3

Graf mehke Petrijeve mreže, ki prikazuje sprožitev akcije t_1 .

7.3 Zvezne Petrijeve mreže

Definicijo zvezne Petrijeve mreže povzemamo po virih [21] in [84]. Zaradi vpeljave časa, ki ga v izrazih označujemo s simbolom t , akcije pri zvezni Petrijevi mreži zaradi nedvoumnosti izrazov označujemo s simbolom t_r .

Definicija 6: Zvezna Petrijeva mreža je definirana s peterico $CPN = (P, T, f, v)$:

- $P = \{p_1, p_2, \dots, p_n\}$ je končna množica vseh pogojev, $n \geq 0$.
- $T = \{t_{r_1}, t_{r_2}, \dots, t_{r_k}\}$ je končna množica vseh akcij, $k \geq 0$.
- Velja $P \cap T = \emptyset$.
- $f : ((P \times T) \cup (T \times P)) \rightarrow \mathbb{R}_0^+$ je funkcija, ki množici usmerjenih povezav določa uteži z nenegativnimi realnimi vrednostmi.
- $v : T \rightarrow H$ je funkcija, ki vsaki akciji t_r določi prožilno funkcijo h_{t_r} , pri čemer je $H := \bigcup_{t_r \in T} \{h_{t_r} | h_{t_r} : \mathbb{R}^{|\bullet t_r|} \rightarrow \mathbb{R}\}$ množica vseh prožilnih funkcij h_{t_r} in velja $v(t_r) = h_{t_r}$.

Označitev pogojev v zvezni Petrijevi mreži je podana z vektorjem $\mathbf{m} = (m(p_1), m(p_2), \dots, m(p_{|P|}))$, pri čemer z $m(p_i) \in (\mathbb{R}_0^+)$, $i = 1, \dots, n$, podamo označitev pogoja p_i , ki je predstavljena z realnim številom. Označitev pogoja ni podana s številom žetonov, ampak z realno vrednostjo, ki ji pravimo *vrednost žetona* (angl. *token value*). Izvajanje zvezne Petrijeve mreže je nenehno. Prožilna funkcija $h_{t_r}(\mathbf{m})$ določa zvezno hitrost

proženja akcije t_r (angl. *continuous transition rate*), njena vrednost pa je odvisna od trenutne označitve Petrijeve mreže ali, natančneje, od trenutne označitve pogojev v množici $\bullet t_r$. Akcija t_r je v zvezni Petrijevi mreži omogočena, če velja $\forall p \in \bullet t_r : m(p) > 0$. Akcije, ki so omogočene, se v zvezni Petrijevi mreži prožijo hipno in nenehno. Za vsak pogoj opazujemo *dinamično ravnovesje* (angl. *dynamic balance*) $B(p, \mathbf{m})$, ki ga zapišemo z izrazom

$$\forall p \in P : B(p, \mathbf{m}) = \sum_{t_r \in \bullet p} f(t_r, p) h_{t_r}(\mathbf{m}) - \sum_{t_r \in p \bullet} f(p, t_r) h_{t_r}(\mathbf{m}). \quad (7.16)$$

Spremembe označitve posameznega pogoja opazujemo v odvisnosti od časa. Spremembe označitve pogoja p_i izračunamo na podlagi izraza

$$m_{t+dt}(p_i) = m_t(p_i) + B_i(p_i, \mathbf{m}_t) * dt, \quad (7.17)$$

kjer $\mathbf{m}_t = (m_t(p_1), m_t(p_2), \dots, m_t(p_{|P|}))$ predstavlja trenutno označitev Petrijeve mreže, $m_t(p_i)$ predstavlja označitev, $B_i(p_i, \mathbf{m}_t)$ pa dinamično ravnovesje pogoja p_i v časovni točki t . Na podlagi združitve izrazov (7.16) in (7.17) lahko izpeljemo diferencialno enačbo, ki jo zapišemo z izrazom

$$\frac{dm(p_i)}{dt} = \sum_{t_r \in \bullet p_i} f(t_r, p_i) h_{t_r}(\mathbf{m}_t) - \sum_{t_r \in p_i \bullet} f(p_i, t_r) h_{t_r}(\mathbf{m}_t), \quad (7.18)$$

kjer $\frac{dm(p_i)}{dt}$ predstavlja spremembo vrednosti žetona v pogoju p_i po času, $h_{t_r}(\mathbf{m}_t)$ prožilno funkcijo akcije t_r , katere vrednost je odvisna od trenutne označitve \mathbf{m}_t , $f(t_r, p_i)$ predstavlja uteži povezav iz akcij, pri katerih pogoj p_i nastopa kot izstopni, in $f(p_i, t_r)$ uteži povezav do akcij, pri katerih pogoj p_i nastopa kot vstopni.

7.4 Uporaba Petrijevih mrež za modeliranje bioloških sistemov

Petrieve mreže so se v zadnjih dveh desetletjih uveljavile tudi za modeliranje kemijskih reakcij in procesov v bioloških sistemih. Za potrebe modeliranja na področju ved o življenju se uporabljajo osnovne in razširjene Petrijeve mreže (zvezne, mehke, stohastične), ki na račun zahtevnosti simulacije bolj natančno opišejo procese in interakcije v bioloških sistemih.

Prvič so se uporabile za opis najosnovnejših biokemijskih reakcij [85, 86], kasneje pa se je njihova uporaba razširila tudi v bolj specifične namene, npr. za opis signalnih poti [87], metabolnih mrež [88] in gensko regulatornih omrežij [89, 90]. Osnovna definicija Petrijevih mrež se pogosto uporablja kot orodje za predstavitev kvalitativnih modelov bioloških sistemov, ki jih modeliramo z Boolovimi mrežami [91]. Petrijeve mreže se uporabljajo tudi za predstavitev nadgradnje Boolovih mrež, t.i. več-vrednostnih mrež, ki se uporabljajo za opis bolj specifičnih kvalitativnih lastnosti bioloških omrežij [31, 92, 93].

Za pol-kvantitativne in kvantitativne opise bioloških procesov so se uveljavile različne razširjene definicije Petrijevih mrež. Hibridne funkcijske Petrijeve mreže so primer razširjenih Petrijevih mrež, ki izkoriščajo prednosti tako kvalitativnega kot tudi kvantitativnega modeliranja in nam omogočajo deloma kvalitativni in deloma kvantitativni opis procesov [89, 94]. Povsem kvantitativni opis bioloških procesov nam omogočajo zvezne in stohastične Petrijeve mreže. S slednjimi lahko modeliramo stohastičnost kemijskih reakcij in dinamiko posameznih molekul v biološkem sistemu [95]. Stohastične Petrijeve mreže se uporabljajo tudi pri analizi in oceni kinetičnih parametrov na podlagi eksperimentalnih podatkov [96]. Stanje biološkega sistema pri determinističnih modelih podamo z vektorjem koncentracij kemijskih zvrsti, ki so predstavljene z realnimi števili. V tem primeru za modeliranje uporabljamo zvezne Petrijeve mreže, ki ustrezajo sistemu diferencialnih enačb [97].

Obstoječe mehke Petrijeve mreže se uporabljajo predvsem pri odkrivanju zakonitosti iz eksperimentalnih podatkov [69], pri odkrivanju različnih interakcij v omrežju genov [16] in za analizo vpliva neznanih vrednosti kinetičnih parametrov pri stohastičnem modeliranju bioloških sistemov [98]. V viru [99] avtorji sicer definirajo mehke zvezne Petrijeve mreže, s katerimi za proženje akcije izberejo eno od znanih vrednosti kinetičnih parametrov, a za inferenco ne definirajo mehkih spremenljivk in mehkih IF-THEN pravil. V procesu odločanja o izbiri vrednosti parametra ne uporabljajo mehke logike, ampak odločanje vršijo na podlagi ostrih vrednosti vhodnih spremenljivk.

7.5 Pomankljivosti obstoječih definicij Petrijevih mrež

Stanje biološkega sistema pri determinističnih modelih predstavimo z vektorjem koncentracij opazovanih kemijskih zvrsti, ki so predstavljene z realnimi pozitivnimi števili, označitev pogoja v osnovnih Petrijevih mrežah pa je predstavljena z diskretnim številom žetonov, zato osnovne Petrijeve mreže niso primerne za kvantitativno modeliranje

enostavnih bioloških sistemov.

Podobno velja tudi za mehke Petrijeve mreže, pri katerih je označitev posameznega pogoja predstavljena z vrednostmi na intervalu $[0, 1]$. Obstoječe definicije mehkih Petrijevih mrež so nezdružljive z ostalimi definicijami, zato je njihova uporaba omejena na sisteme, ki jih modeliramo izključno z mehko logiko.

Zvezne Petrijeve mreže za proženje akcij in izračun nove označitve potrebujejo natančne vrednosti kinetičnih parametrov, ki pogojujejo dinamiko sistema in se nahajajo bodisi v utežeh povezav bodisi v prožilnih funkcijah h_t . V primeru, da vrednosti kinetičnih parametrov oziroma eksperimentalni podatki, na podlagi katerih bi vrednosti kinetičnih parametrov ocenili, niso na voljo, simulacija z zveznimi Petrijevimi mrežami ni mogoča.

V nadaljevanju predstavimo definicijo zveznih Petrijevih mrež z mehko logiko, ki nam omogočajo modeliranje enostavnih bioloških sistemov, za katere določene vrednosti kinetičnih parametrov niso znane, in so združljive z obstoječimi zveznimi Petrijevimi mrežami.

7.6 Zvezne Petrijeve mreže z mehko logiko

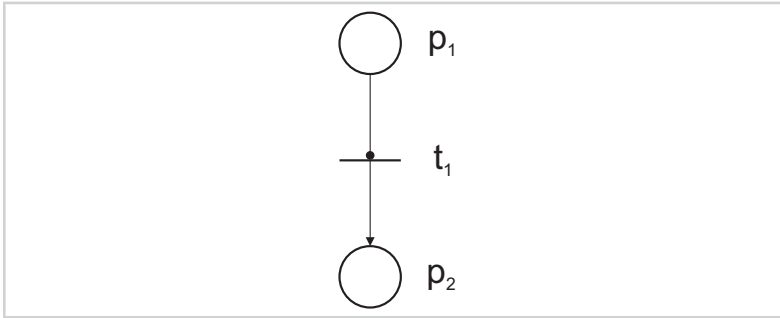
Kot izhodišče za razširitev z mehko logiko uporabimo definicijo zveznih Petrijevih mrež. Pri mehkem opisu procesa v pogojih Petrijeve mreže uporabljamo zvezne vrednosti iz intervala $[0, \infty]$. Obstoječo definicijo zveznih Petrijevih mrež razširimo z naborom *mehkih prožilnih funkcij*, ki v akcijah zvezne Petrijeve mreže z mehko logiko predstavljajo mehki opis procesa. Razširjena definicija predstavlja izvirni znanstveni doprinos pričujoče disertacije.

Definicija 7: Zvezne Petrijeve mreže z mehko logiko definiramo s šesterico $CPNFL = (P, T, f, v, v_{FL}, m_0)$:

- $P = \{p_1, p_2, \dots, p_n\}$ je končna množica vseh pogojev, $n \geq 0$.
- $T = \{t_{r_1}, t_{r_2}, \dots, t_{r_k}\}$ je končna množica vseh akcij, $k \geq 0$.
- Velja $P \cap T = \emptyset$.
- $f : ((P \times T) \cup (T \times P)) \rightarrow \mathbb{R}_0^+$ je funkcija, ki množici usmerjenih povezav določa uteži z nenegativnimi realnimi vrednostmi.

- $v : T \rightarrow H$ je funkcija, ki vsaki akciji t_r določi prožilno funkcijo h_{t_r} , pri čemer je $H := \bigcup_{t_r \in T} \{h_{t_r} | h_{t_r} : \mathbb{R}^{|\bullet t_r|} \rightarrow \mathbb{R}\}$ množica vseh prožilnih funkcij h_{t_r} in velja $v(t_r) = h_{t_r}$.
- $v_{FL} : T \rightarrow H_{FL}$ je funkcija, ki akciji t_r določi mehko prožilno funkcijo $h_{tr_{FL}}$, pri čemer je $H_{FL} := \bigcup_{tr_{FL} \in T} \{h_{tr_{FL}} | h_{tr_{FL}} : \mathbb{R}^{|\bullet t_r|} \rightarrow \mathbb{R}\}$ množica vseh mehkih prožilnih funkcij in velja $v_{FL}(t) = h_{tr_{FL}}$ za vse akcije $t_r \in T$, ki predstavljajo mehko opisan proces.

Množica prožilnih funkcij H_{FL} vsebuje mehke prožilne funkcije $h_{tr_{FL}}$ vseh akcij v zvezni Petrijevi mreži z mehko logiko, ki predstavljajo procese z neznanimi vrednostmi kinetičnih parametrov. Za potrebe izvajanja zvezne Petrijeve mreže z mehko logiko uporabljamo tudi bralne povezave med pogojem in akcijo. Bralna povezava pri proženju akcije ne zmanjša vrednosti žetona v vhodnem pogoju. V grafu zvezne Petrijeve mreže z mehko logiko pri bralnih povezavah namesto puščice narišemo poln krog kot je prikazano na sliki 7.4.



Slika 7.4

Primer bralne povezave med pogojem p_1 in akcijo t_1 . V primeru bralne povezave se pri proženju akcije vrednosti žetona v vhodnem pogoju p_1 ne zmanjša.

Izvajanje zvezne Petrijeve mreže z mehko logiko poteka podobno kot izvajanje navadne zvezne Petrijeve mreže. Diferencialno enačbo (7.18), ki določa spremembo označitve posameznega pogoja v zvezni Petrijevi mreži, lahko zapišemo v obliki, ki nam podaja tudi odvisnost prožilnih funkcij od vektorja vrednosti kinetičnih parametrov σ . Zapišemo jo z izrazom

$$\frac{dm(p_i)}{dt} = \sum_{j=1}^l g_{ij}(\mathbf{m}_t, \boldsymbol{\sigma}), \quad i = 1, \dots, n, \quad l = |\bullet p_i \cup p_i^\bullet|, \quad (7.19)$$

kjer $\frac{dm(p_i)}{dt}$ predstavlja spremembo označitve pogoja p_i po času, $g_{ij}(\mathbf{m}_t, \boldsymbol{\sigma})$ pa funkcije, katerih vrednost izračunamo na podlagi prožilnih funkcij $h_{t_r}(\mathbf{m}_t)$, vrednosti uteži vhodnih in izhodnih povezav pogoja p_i in vektorja vrednosti kinetičnih parametrov $\boldsymbol{\sigma}$, ki nastopajo v funkcijah $h_{t_r}(\mathbf{m}_t)$. Funkcija $g_i(\mathbf{m}_t, \boldsymbol{\sigma})$ je torej sestavljena in jo za vhodne pogoje $p_i \in \bullet t_{r_j}$ akcije t_{r_j} zapišemo z izrazom

$$g_{IN_{ij}}(\mathbf{m}_t, \boldsymbol{\sigma}) = f(p_i, t_{r_j}) h_{t_{r_j}}(\mathbf{m}_t, \boldsymbol{\sigma}), \quad (7.20)$$

za izhodne pogoje $p_i \in t_{r_j}^\bullet$ akcije t_{r_j} pa z izrazom

$$g_{OUT_{ji}}(\mathbf{m}_t, \boldsymbol{\sigma}) = f(t_{r_j}, p_i) h_{t_{r_j}}(\mathbf{m}_t, \boldsymbol{\sigma}). \quad (7.21)$$

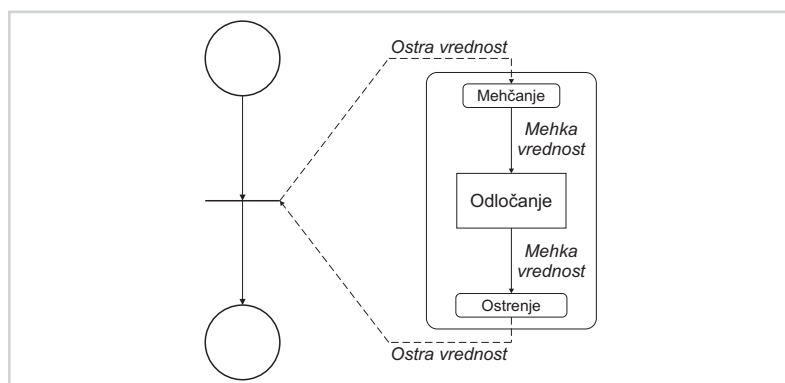
Vrednost funkcije $g(\mathbf{m}_t, \boldsymbol{\sigma})$ je lahko negativna. Na grafu zvezne Petrijeve mreže funkcije $g(\mathbf{m}_t, \boldsymbol{\sigma})$ zapišemo na povezave. Če je povezava med vhodnim pogojem p_i in akcijo t_{r_j} bralna, velja $h_{t_{r_j}}(\mathbf{m}_t) = 0$ in posledično $g_{IN_{ij}}(\mathbf{m}_t, \boldsymbol{\sigma}) = 0$. V primeru, da za katerokoli od prožilnih funkcij vrednost enega od kinetičnih parametrov iz $\boldsymbol{\sigma}$ ni znana, funkcije in posledično proženja akcije ni mogoče ovrednotiti. Takšne prožilne funkcije zamenjamo z *mehkimi prožilnimi funkcijami* in tako pridemo do izraza

$$\frac{dm(p_i)}{dt} = \sum_{s=1}^q FL_{i,s}(\mathbf{m}_t) + \sum_{j=1}^r g_{ij}(\mathbf{m}_t, \boldsymbol{\sigma}^*), \quad q + r = |\bullet p_i \cup p_i^\bullet|, \quad i = 1, \dots, n, \quad (7.22)$$

kjer $FL_{i,s}(\mathbf{m}_t)$ predstavljajo funkcije, ki opisujejo spremembe označitve vhodnih in izhodnih pogojev akcij, za katere vrednosti kinetičnih parametrov niso znane, $\boldsymbol{\sigma}^*$ pa vektor parametrov z znanimi vrednostmi, pri čemer velja $|\boldsymbol{\sigma}^*| < |\boldsymbol{\sigma}|$. Njihovo vrednost izračunamo na podlagi trenutne označitve \mathbf{m}_t in za proces ustrezne mehke prožilne funkcije $h_{tr_{FL}}(\mathbf{m}_t)$. Podobno kot pri funkcijah $g(\mathbf{m}_t, \boldsymbol{\sigma})$ tudi pri mehkih prožilnih funkcijah velja $h_{tr_{FL}}(\mathbf{m}_t) = 0$ in posledično $FL_{i,s}(\mathbf{m}_t) = 0$, kadar je povezava med vhodnim pogojem in akcijo bralna.

7.7 Modeliranje bioloških procesov z zveznimi Petrijevim mrežami z mehko logiko

Mehke prožilne funkcije posameznega procesa določimo s postopkom, opisanim v poglavju 6. V vsakem časovnem intervalu dt se prožijo vse akcije, spremembe vrednosti žetonov posameznih pogojev pa se izračunajo na podlagi izraza (7.22). Za mehko prožilno funkcijo velja, da je proženje akcije, ki ji funkcija pripada, sestavljeno iz treh korakov, in sicer izračuna mehke vrednosti vhodnih pogojev, odločanja na podlagi mehkih IF-THEN pravil ter ostrenja izhodne vrednosti. Tri korake akcije, ki ji pripada mehka prožilna funkcija, opisuje shema na sliki 7.5.



Slika 7.5

Proženje akcije, ki ji pripada mehka prožilna funkcija. Akcija z enim proženjem izvede tri korake: pridobivanje mehke vrednosti na vходу, odločanje na osnovi mehkih IF-THEN pravil in izračun ostre vrednosti na izhodu.

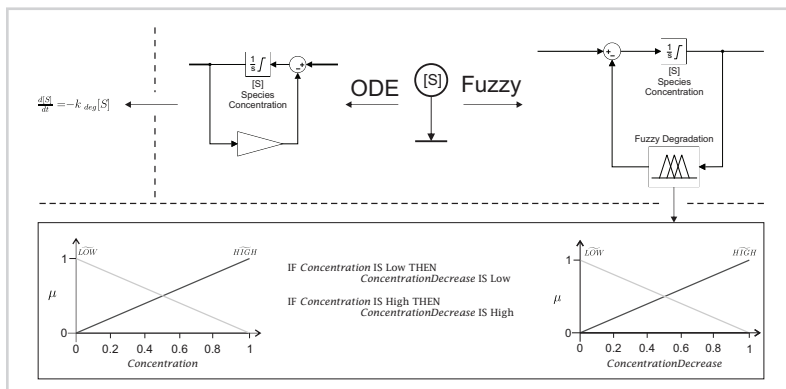
V nadaljevanju definiramo mehke opise osnovnih bioloških procesov gensko regulatornih omrežij, ki se v zvezni Petrijevi mreži z mehko logiko uporabljajo kot mehke prožilne funkcije. Izhodi mehkih prožilnih funkcij so ostre vrednosti sprememb koncentracij. Slednje nam omogoča integracijo mehkih prožilnih funkcij v obstoječo definicijo zveznih Petrijevih mrež, s čimer ohranimo interoperabilnost obeh vrst prožilnih funkcij. Ker se vrednosti maksimalnih koncentracij in maksimalnih sprememb koncentracij za procese v različnih sistemih lahko razlikujejo, na grafih, ki prikazujejo pripadnostne funkcije vhodnih in izhodnih spremenljivk, uporabljamo vrednosti normalizirane na interval $[0, 1]$.

7.7.1 Mehka prožilna funkcija za modeliranje degradacije

Prispevek k spremembi stanja sistema zaradi procesa degradacije v diferencialni enačbi (7.19) kot funkcijo zapišemo z izrazom

$$\frac{d[S]}{dt} = -k_{deg}[S], \quad (7.23)$$

kjer je $[S]$ trenutna koncentracija opazovane kemijske zvrsti, k_{deg} koeficient degradacije in $\frac{d[S]}{dt}$ sprememba koncentracije opazovane kemijske zvrsti po času. Prožilna funkcija v zvezni Petrijevi mreži, ki opisuje proces degradacije, je odvisna od vrednosti vhodnega pogoja, ki predstavlja trenutno koncentracijo opazovane kemijske zvrsti $[S]$, in od vrednosti konstante k_{deg} . V primeru, da slednja ni znana, degradacijo opišemo z akcijo, ki ima en vhodni pogoj $[S]$ in ni odvisna od konstante k_{deg} . Velikost spremembe koncentracije zaradi procesa degradacije glede na enačbo (7.23) z večanjem koncentracije konstantno linearno narašča. Takšni dinamiki ustrezne pripadnostne funkcije vhodne in izhodne mehke spremenljivke ter mehka IF-THEN pravila prikazuje spodnji del slike 7.6.



Slika 7.6

Primerjava implementacije akcije (prožilnih funkcij) za proces degradacije z diferencialno enačbo in z mehkim opisom s pripadajočimi shemami v orodju MATLAB Simulink® (zgoraj). Pripadnostne funkcije vhodne in izhodne spremenljivke ter mehka IF-THEN pravila za mehki opis (spodaj).

V določenih primerih degradacijo modeliramo kot encimsko reakcijo. V tem primeru prispevek k spremembi stanja sistema kot funkcijo zapišemo z izrazom

$$\frac{d[S]}{dt} = \frac{k_{deg}[S]}{1 + \frac{[S]}{K_m}} \quad (7.24)$$

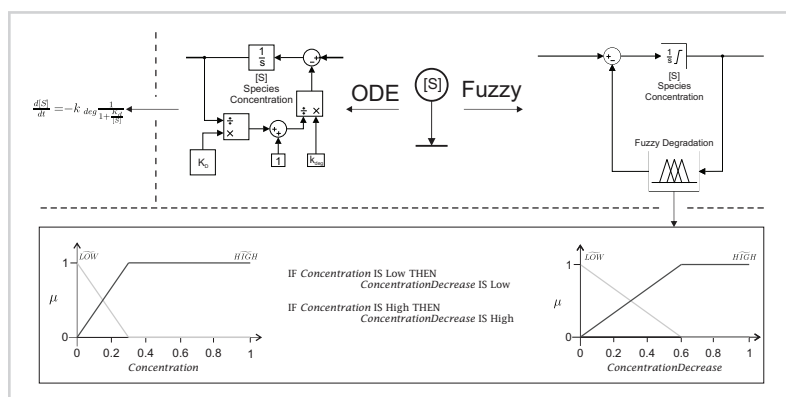
$\frac{1}{s}$ Species Concentration K_m 1 k_{deg} $\frac{1}{s}$ Species Concentration Fuzzy Degradation

IF Concentration IS Low THEN ConcentrationDecrease IS Low
 IF Concentration IS High THEN ConcentrationDecrease IS High

1
 0 0.2 0.4 0.6 0.8 1
 Concentration

1
 0 0.2 0.4 0.6 0.8 1
 ConcentrationDecrease

kjer je $[S]$ trenutna koncentracija opazovane kemijske zvrsti, k_{deg} koeficient degradacije, K_d Michaelisova konstanta in $\frac{d[S]}{dt}$ sprememba koncentracije opazovane kemijske zvrsti po času. V primeru, da vrednosti kinetičnih parametrov k_{deg} ali K_d nista znani, proces encimske degradacije opišemo z akcijo, ki ima en vhodni pogoj $[S]$ in ni odvisna od nobene konstante. Pri določitvi mehke prožilne funkcije je potrebno upoštevati posebnosti encimske degradacije. V primeru slednje pri visokih koncentracijah opazovane kemijske zvrsti pride do nasičenja. Od točke, ki predstavlja nasičenje koncentracije, se sprememba koncentracije zaradi degradacije ne povečuje več. Približno poznavanje vrednosti koncentracije, pri kateri pride do nasičenja, uporabimo pri določanju pripadnostnih funkcij vhodne in izhodne mehke spremenljivke. Obliko mehkih spremenljivk in mehkih IF-THEN pravil za proces encimske degradacije prikazuje spodnji del slike 7.7.



Slika 7.7

Primerjava implementacije akcije (prožilnih funkcij) za proces encimske degradacije z diferencialno enačbo in z mehkim opisom s pripadajočimi shemami v orodju MATLAB Simulink® (zgoraj). Pripadnostne funkcije vhodne in izhodne spremenljivke ter mehka IF-THEN pravila za mehki opis (spodaj). V primeru mehkega opisa prikazanega na sliki predpostavljamo, da je točka nasičenja pri približno tretjini maksimalne koncentracije (glej pripadnostne funkcije vhodne spremenljivke).

Pri procesu degradacije se vrednost žetona v pogoju $[S]$ zmanjša, ne vpliva pa na spremembo vrednosti drugih pogojev v Petrijevi mreži. Izvedba akcije, ki opisuje proces degradacije, povzroči zmanjšanje vrednosti vhodnega pogoja $[S]$. Velikost slednjega je enaka izhodni spremenljivki mehke prožilne funkcije degradacije.

7.7.2 Mehka prožilna funkcija za modeliranje translacije

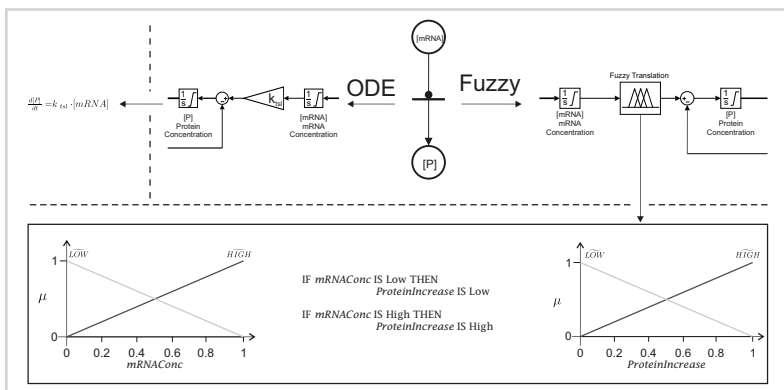
Prispevek k spremembi stanja sistema zaradi procesa translacije v diferencialni enačbi (7.19) kot funkcijo zapišemo z izrazom

$$\frac{d[P]}{dt} = k_{tsl}[mRNA_P], \quad (7.25)$$

kjer je $[mRNA_P]$ trenutna koncentracija mRNA iz katere s procesom translacije nastaja protein $[P]$, k_{tsl} koeficient translacije in $\frac{d[P]}{dt}$ sprememba koncentracije opazovanega proteina po času. Prožilna funkcija v zvezni Petrijevi mreži, ki opisuje proces translacije, je odvisna od vrednosti vhodnega pogoja, ki predstavlja trenutno koncentracijo mRNA, in od konstante k_{tsl} . V primeru, da vrednost k_{tsl} ni znana, translacijo opišemo z akcijo, ki ima en vhodni zvezni pogoj $[mRNA_P]$, en izhodni zvezni pogoj $[P]$ in ni odvisna od konstante k_{tsl} . Povečanje koncentracije proteina pri translaciji konstantno narašča z večanjem koncentracije pripadajoče mRNA. Pripadnostne funkcije vhodne in izhodne mehke spremenljivke ter mehka IF-THEN pravila, ki takšen način večanja spremembe opisujejo, za proces translacije prikazuje spodnji del slike 7.8.

Slika 7.8

Primerjava implementacije akcije (prožilnih funkcij) za proces translacije z diferencialno enačbo in z mehkim opisom s pripadajočimi shemami v orodju MATLAB Simulink® (zgoraj). Pripadnostne funkcije vhodne in izhodne spremenljivke ter mehka IF-THEN pravila, ki opisujejo rast spremembe koncentracije proteina z večanjem koncentracije mRNA zaradi procesa translacije, so predstavljene na spodnjem delu slike.



Pri procesu translacije se vrednost žetona vhodnega pogoja $[mRNA_P]$ ne spremeni, poveča pa se vrednost žetona izhodnega pogoja $[P]$. Spremembo v označitvi Petrijeve mreže izračunamo na podlagi prebrane vrednosti vhodnega pogoja in mehke prožilne funkcije, ki opisuje proces translacije.

7.7.3 Mehka prožilna funkcija za modeliranje transkripcije

V poglavju 2 smo predstavili vpliv transkripcijskih faktorjev na izražanje gena. Slednji lahko izražanje gena zavirajo (represija) ali pospešijo (aktivacija). Podobno kot pri

obtoječih metodah na osnovi diferencialnih enačb ločimo mehka opisa transkripcije z represijo in transkripcije z aktivacijo.

Represija izražanja

Prispevek k spremembi stanja sistema zaradi procesa transkripcije v primeru represije v diferencialni enačbi (7.19) kot funkcijo zapišemo z izrazom

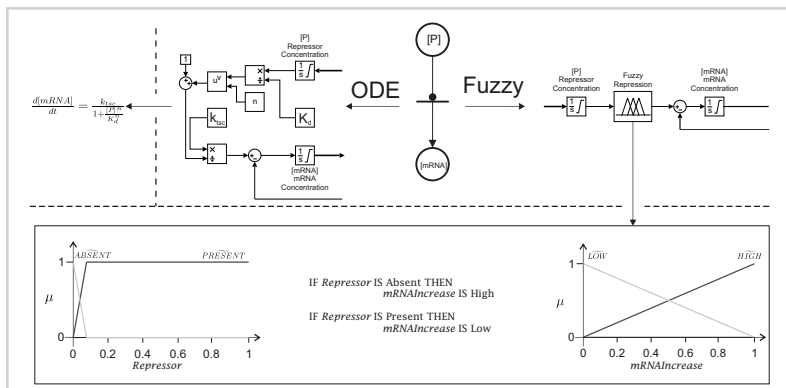
$$\frac{d[mRNA]}{dt} = k_{tsc} \cdot \frac{1}{1 + \frac{[P]^n}{K_d^n}}, \quad (7.26)$$

kjer je $[P]$ trenutna koncentracija proteina, ki v procesu nastopa kot represor, K_d disociacijska konstanta represorja in njegovega vezavnega mesta, k_{tsc} koeficient transkripcije, n Hillov koeficient in $\frac{d[mRNA]}{dt}$ sprememba koncentracije opazovane mRNA po času. Prožilna funkcija v zvezni Petrijevi mreži, ki opisuje transkripcijo, je odvisna od vrednosti vhodnega pogoja $[P]$, ki predstavlja trenutno koncentracijo represorja ter od vrednosti konstant k_{tsc} , K_d in n . V primeru, da slednje niso znane, proces transkripcije opišemo z mehko prožilno funkcijo, ki ima en vhodni zvezni pogoj $[P]$ in en izhodni zvezni pogoj $[mRNA]$. Lastnosti mehke prožilne funkcije določimo v skladu s poznavanjem vplivov posameznih kinetičnih parametrov na dinamiko procesa. Moč vezave represorja na promotor pri danem transkripcijskem faktorju, ki ga opisuje kinetični parameter K_d , je določena s koncentracijo represorja, pri kateri je izražanje gena zaradi represije polovično. Približno poznavanje intenzivnosti represorja nam tako omogoča določitev točke v kateri se pripadnostni funkciji, ki opisujeta prisotnost oziroma odsotnost represorja, sekata. Sečna točka blizu ničle ustreza represorju z zelo močno vezavo, medtem ko sečna točka daleč od ničle ustreza represorju s šibko vezavo. Pripadnostne funkcije vhodne in izhodne mehke spremenljivke ter mehka IF-THEN pravila za proces transkripcije z represijo, ki določajo opisano dinamiko procesa, prikazuje spodnji del slike 7.9.

Pri represiji se vrednost žetona vhodnega pogoja $[P]$ ne spremeni, poveča pa se vrednost žetona izhodnega pogoja $[mRNA]$. Spremembo v označitvi Petrijeve mreže izračunamo na podlagi prebrane vrednosti žetona vhodnega pogoja in mehke prožilne funkcije represije.

Slika 7.9

Primerjava implementacije akcije (prožilnih funkcij) za proces transkripcije (represija) z diferencialno enačbo in z mehkim opisom s pripadajočimi shemami v orodju MATLAB Simulink (zgoraj). Pripadnostne funkcije vhodne in izhodne spremenljivke ter mehka IF-THEN pravila za mehki opis so predstavljene na spodnjem delu sheme. V primeru mehkega opisa prikazanega na sliki predpostavljamo, da je sečna točka pri relativno nizki koncentraciji represorja (blizu ničle). Pripadnostne funkcije vhodne spremenljivke torej ustrezajo opisu represorja z močno vezavo.



Aktivacija izražanja

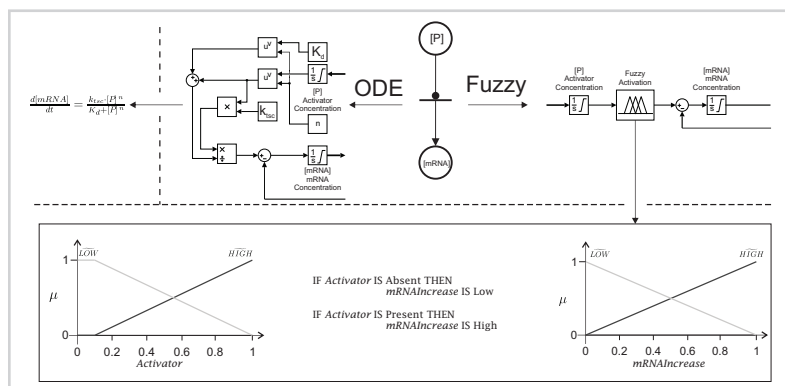
Prispevek k spremembi stanja sistema zaradi procesa transkripcije v primeru aktivacije v diferencialni enačbi (7.19) kot funkcijo zapišemo z izrazom

$$\frac{d[mRNA]}{dt} = k_{tsc} \cdot \frac{[P]^n}{K_d^n + [P]^n}, \quad (7.27)$$

kjer je $[P]$ trenutna koncentracija proteina, ki v procesu nastopa kot aktivator, K_d disociacijska konstanta aktivatorja in njegovega vezavnega mesta, ostale spremenljivke in konstante pa so enake kot pri represiji. Podobno kot pri represiji tudi pri aktivaciji sečna točka pripadnostnih funkcij vhodne spremenljivke predstavlja moč vezave aktivatorja na promotor. Pripadnostne funkcije vhodne in izhodne mehke spremenljivke ter mehka IF-THEN pravila za proces aktivacije izražanja gena prikazuje spodnji del slike 7.10.

Podobno kot pri represiji se tudi pri aktivaciji vrednost žetona vhodnega pogoja $[P]$ ne spremeni, poveča pa se vrednost žetona izhodnega pogoja $[mRNA]$.

Slika 7.10



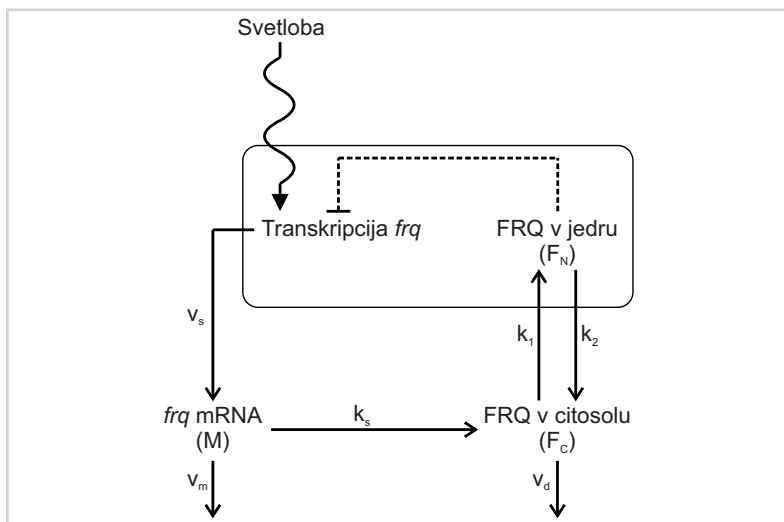
Primerjava implementacije akcije (prožilnih funkcij) za proces transkripcije (aktivacija) z diferencialno enačbo in z mehkim opisom s pripadajočimi shemami v orodju MATLAB Simulink™ (zgoraj). Pripadnostne funkcije vhodne in izhodne spremenljivke ter mehka IF-THEN pravila za mehki opis so predstavljene na spodnjem delu slike. V primeru mehkega opisa prikazanega na sliki predpostavljamo, da je sečna točka pri relativno visoki koncentraciji aktivatorja (daleč od ničle). Pripadnostne funkcije vhodne spremenljivke torej ustrezajo opisu aktivatorja s šibko vezavo.

7.8 Zvezna Petrijeva mreža z mehko logiko za modeliranje cirkadianega ritma glive *Neurospora*

Gliva *Neurospora* služi kot modelni organizem za raziskovanje molekularnih in fizioloških lastnosti cirkadianih ritmov. Na slednje v glivi *Neurospora* vpliva veliko različnih dejavnikov, kot glavna pa se izpostavljata protein imenovan FREQUENCY (FRQ) in svetloba. Osnovni model, ki opisuje cirkadiani ritem in ga povzemamo po viru [100], je sestavljen iz ene negativne povratne zanke. Zanko tvorijo proces transkripcije mRNA, ki se prepiše v protein FRQ, in prehod FRQ iz citosola v jedro, kjer FRQ zavira izražanje gena *frq* (gen proteina FRQ). Model negativne povratne zanke prikazuje shema na sliki 7.11.

7.8.1 Referenčni deterministični model glive *Neurospora*

Osnovni model opisujejo spremembe koncentracij mRNA ter proteina FRQ v citosolu in istega proteina v jedru. Določene so z naslednjim sistemom diferencialnih enačb



Slika 7.11

Shema delovanja cirkadianega ritma v glivi *Neurospora* povzeta po [100]. Gen *frq* se prepiše v mRNA iz katere nastane protein FRQ. S prehodom proteina FRQ v jedro ta zavira izražanje gena *frq*.

$$\frac{d[M]}{dt} = v_s \frac{1}{1 + \frac{[F_N]^n}{K_I^n}} - v_m \frac{1}{1 + \frac{K_m}{[M]}}, \quad (7.28)$$

$$\frac{d[F_C]}{dt} = k_s[M] - v_d \frac{1}{1 + \frac{K_d}{[F_C]}} - k_1[F_C] + k_2[F_N], \quad (7.29)$$

$$\frac{d[F_N]}{dt} = k_1[F_C] - k_2[F_N], \quad (7.30)$$

kjer je $[M]$ koncentracija mRNA, $[F_C]$ koncentracija proteina FRQ v citosolu, $[F_N]$ koncentracija proteina FRQ v jedru celice, v_s koeficient transkripcije, K_I koeficient represije, n Hillov koeficient, v_m koeficient degradacije mRNA, K_m Michaelisova konstanta degradacije za mRNA, k_s koeficient translacije, v_d koeficient degradacije FRQ v citosolu, K_d Michaelisova konstanta degradacije FRQ, k_1 koeficient prehoda FRQ iz citosola v jedro in k_2 koeficient prehoda FRQ iz jedra v citosol. Tabela 7.1 prikazuje vrednosti kinetičnih parametrov, ki smo jih za potrebe pridobivanja simulacijskih rezultatov povzeli po viru [100].

Tabela 7.1

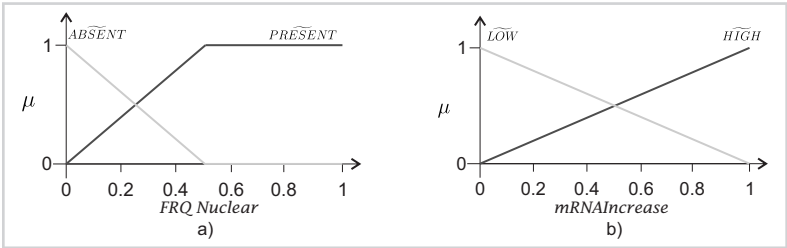
Vrednosti kinetičnih parametrov, ki smo jih uporabili v izhodiščnem referenčnem modelu [100].

<i>n.</i>	<i>Proces</i>	<i>Parameter</i>	<i>Vrednost</i>
(2)	Transkripcija	v_s	1.6 nM/h
(3)	Transkripcija	n	4
(4)	Transkripcija	K_I	1 nM
(5)	Translacija	k_s	0.5 h ⁻¹
(6)	Degradacija mRNA	v_m	0.505 nM/h
(7)	Degradacija mRNA	K_m	0.5 nM
(8)	Degradacija FRQ v citosolu	v_d	1.4 nM/h
(9)	Degradacija FRQ v citosolu	K_d	0.13 nM
(10)	Prehod FRQ iz citosola v jedro	k_1	0.5 h ⁻¹
(11)	Prehod FRQ iz jedra v citosol	k_2	0.6 h ⁻¹

7.8.2 Model z mehкими prožilnimi funkcijami

Za modeliranje glive *Neurospora* z mehko logiko smo za vsakega od procesov definirali pripadajoče mehke prožilne funkcije. Pri vseh prožilnih funkcijah smo uporabili vhodne in izhodne spremenljivke ter mehka IF-THEN pravila, ki smo jih definirali v razdelku 7.7. Za vse procese smo uporabili defuzifikacijsko metodo *povprečna vrednost maksimuma* (angl. *mean of maximum* - MOM).

V mehki prožilni funkciji za transkripcijo nastopata vhodna spremenljivka *FRQ Nuclear*, ki predstavlja koncentracijo proteina FRQ v jedru, in izhodna spremenljivka *mRNAIncrease*, ki predstavlja povečanje koncentracije mRNA. Pripadnostne funkcije za vhodno in izhodno spremenljivko prikazuje slika 7.12.



Slika 7.12

Pripadnostne funkcije vhodne spremenljivke, ki predstavlja koncentracijo proteina FRQ v jedru (a), in izhodne spremenljivke, ki predstavlja povečanje koncentracije mRNA (b).

Pripadnostne funkcije lahko podamo tudi v parametričnem zapisu, pri čemer z zapisom $\mu_{\bar{x}} = [a, b, c]_p$ podajamo pripadnostno funkcijo trikotne oblike, z zapisom $\mu_{\bar{y}} = [a, b, c, d]_p$ pa pripadnostno funkcijo s trapezoidno obliko (glej razdelek 4.2). Parametrični zapis pripadnostnih funkcij, ki smo jih definirali za vhodno spremenljivko *FRQNuclear*, podamo z izrazom

$$\mu_{\widetilde{\text{ABSENT}}} = [0, 0, 0.5]_p, \mu_{\widetilde{\text{PRESENT}}} = [0, 0.5, 1, 1]_p, \quad (7.31)$$

za izhodno spremenljivko *mRNAIncrease* pa z izrazom

$$\mu_{\widetilde{\text{LOW}}} = [0, 0, 1]_p, \mu_{\widetilde{\text{HIGH}}} = [0, 1, 1]_p. \quad (7.32)$$

V mehki prožilni funkciji za translacijo nastopata vhodna spremenljivka *mRNAConc*, ki predstavlja koncentracijo mRNA, in izhodna spremenljivka *FRQCytosolicIncrease*, ki predstavlja povečanje koncentracije proteina FRQ v citosolu. Parametrični zapis pripadnostnih funkcij, ki smo jih definirali za vhodno spremenljivko *mRNAConc*, podamo z izrazom

$$\mu_{\widetilde{\text{LOW}}} = [0, 0, 1]_p, \mu_{\widetilde{\text{HIGH}}} = [0, 1, 1]_p, \quad (7.33)$$

za izhodno spremenljivko *FRQCytosolicIncrease* pa z izrazom

$$\mu_{\widetilde{\text{LOW}}} = [0, 0, 1]_p, \mu_{\widetilde{\text{HIGH}}} = [0, 1, 1]_p. \quad (7.34)$$

V mehkih prožilnih funkcijah za degradacijo proteina FRQ in mRNA nastopata vhodni spremenljivki *FRQCytosolicConc* in *mRNAConc*, ki predstavljata koncentraciji proteina FRQ v citosolu ter mRNA, in izhodni spremenljivki *FRQCytosolicDecrease* in *mRNADecrease*, ki predstavljata zmanjšanje koncentracije proteina FRQ v citosolu ter mRNA. Parametrični zapis pripadnostnih funkcij, ki smo jih definirali za vhodni spremenljivki *FRQCytosolicConc* in *mRNAConc*, podamo z izrazom

$$\mu_{\widetilde{\text{LOW}}} = [0, 0, 0.3]_p, \mu_{\widetilde{\text{HIGH}}} = [0, 0.3, 1, 1]_p, \quad (7.35)$$

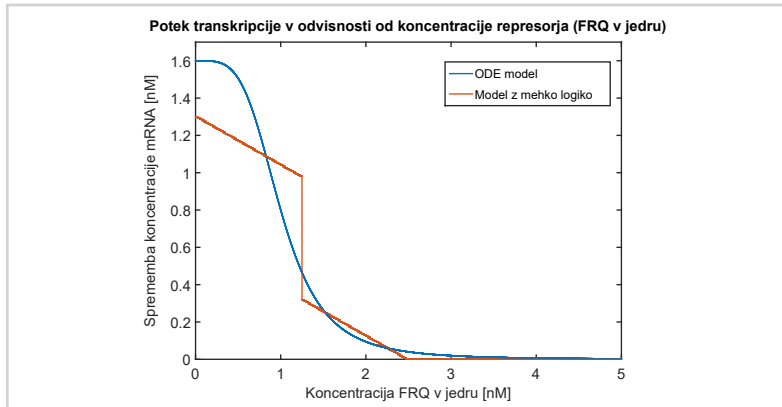
za izhodni spremenljivki *FRQ_{CytosolicDecrease}* in *mRNA_{Decrease}* pa z izrazom

$$\mu_{\text{LOW}} = [0, 0, 0.6]_p, \mu_{\text{HIGH}} = [0, 0.6, 1, 1]_p. \quad (7.36)$$

Kor primer odzivov sta na sliki 7.13 prikazana odziva izhodne spremenljivke na različne vrednosti vhodne spremenljivke referenčnega in mehkega modela procesa transkripcije. Referenčni model transkripcije je zgrajen na podlagi izraza

$$f([F_N]) = v_s \frac{1}{1 + \frac{[F_N]^n}{K_I^n}}, \quad (7.37)$$

kjer $[F_N]$ predstavlja koncentracijo proteina FRQ v jedru, parametri v_s , n in K_I pa predstavljajo hitrost transkripcije, Hillov koeficient in koeficient transkripcije, katerih vrednosti so navedene v tabeli 7.1. Pri mehkem modelu uporabljamo pripadnostne funkcije, ki so prikazane na sliki 7.12, pri čemer so vrednosti vhodne spremenljivke iz intervala $[0, 5]$ normalizirane na interval $[0, 1]$, vrednosti izhodne spremenljivke pa iz intervala $[0, 1]$ na interval $[0, 1.3]$.



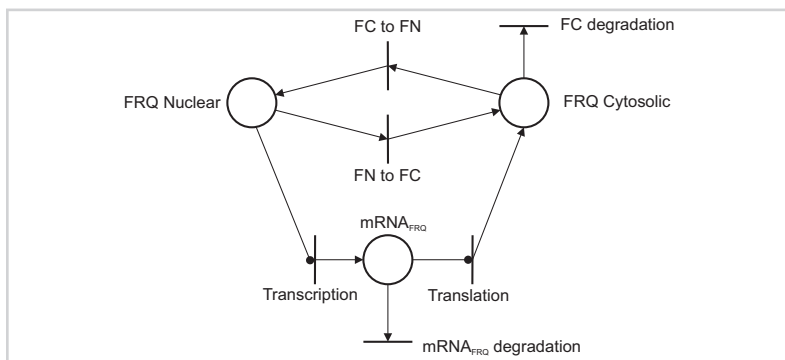
Slika 7.13

Vpliv represorja na transkripcijo. Z rdečo je označena sprememba koncentracije mRNA pri različnih koncentracijah represorja (protein FRQ v jedru) za mehki model transkripcije, z modro pa za model na osnovi diferencialne enačbe. Pri mehkem modelu so vrednosti vhodne spremenljivke pred vhomom v mehki sistem iz intervala $[0, 5]$ normalizirane na interval $[0, 1]$, vrednosti izhodne spremenljivke po izhodu iz mehkega sistema preslikane iz intervala $[0, 1]$ na interval $[0, 1.3]$.

Graf Petrijeve mreže, ki ustreza osnovnemu modelu cirkadianega ritma glive *Neurospora*, je prikazana na sliki 7.14.

Slika 7.14

Graf Petrijeve mreže, ki opisuje delovanje cirkadianega ritma glive *Neurospora*. V akcijah se nahajajo prožilne funkcije, ki opisujejo delovanje posameznega procesa. Prožilne funkcije akcij v primeru nepoznavanja vrednosti kinetičnih parametrov za posamezen proces nadomestimo z mehкими prožilnimi funkcijami.

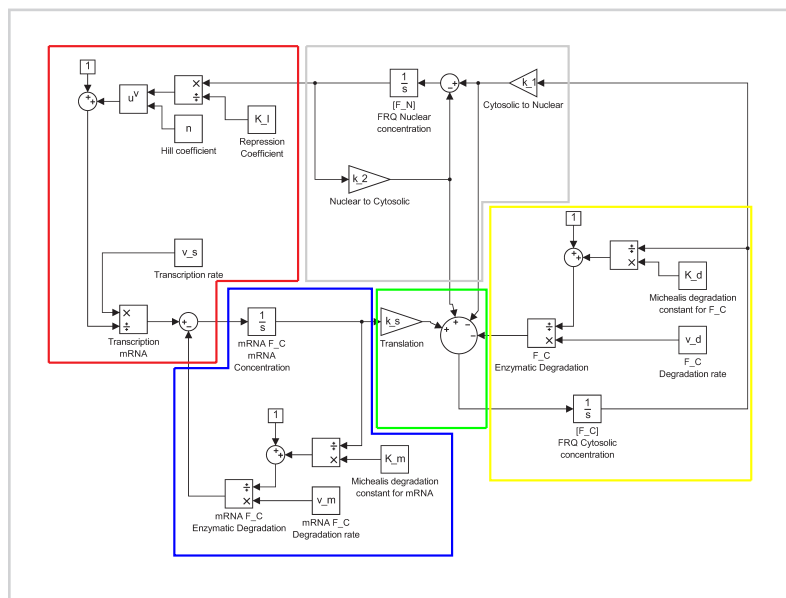


7.8.3 Uporaba mehkih prožilnih funkcij za modeliranje bioloških procesov glive *Neurospora*

Tako referenčni model kot tudi mešane mehko-deterministične modele smo zgradili v programskem okolju MATLAB Simulink® [51]. Pri mehko-determinističnih modelih smo posamezne procese zamenjali z njihovimi mehкими opisi, ki smo jih predstavili v razdelku 7.8.2. Diagram referenčnega modela prikazuje slika 7.15, diagram modela, pri katerem smo transkripcijo opisali z mehкim modelom, pa 7.16.

Vse simulacije so potekale pod istimi pogoji, in sicer smo za numerično reševanje uporabljali simulacijski pogon *ode4* (*Runge-Kutta*) z nespremenljivim časovnim korakom 0.01 h. Kadar ni navedeno drugače, smo uporabljali vrednosti kinetičnih parametrov iz tabele 7.1, ki so povzeti po [100]. Simulirali smo 72 ur dinamike sistema in opazovali amplitudo in frekvenco cirkadianega ritma. Simulacijski rezultati vseh modelov so prikazani na sliki 7.17.

Iz simulacijskih rezultatov je razvidno, da zamenjava prožilnih funkcij, ki temeljijo na diferencialnih enačbah, z mehкими prožilnimi funkcijami bistveno ne vpliva na frekvenco in amplitudo cirkadianega ritma. Izrazito se frekvenca spremeni le v primeru na sliki 7.17h, pri katerem smo vse prožilne funkcije zamenjali z mehкими prožilnimi funkcijami. Poleg razlike v frekvenci so pri slednjem oscilacije tudi dušene, medtem ko pri vseh ostalih modelih koncentracije oscilirajo stabilno. Pri vsakem od bioloških procesov smo faktorje funkcij *prescale* in *postscale* za preslikavo vhoda na interval [0, 1] in izhoda iz intervala [0, 1] za mehке prožilne funkcije nastavili na podlagi približnega poznavanja maksimalnih koncentracij in sprememb koncentracij, ki jih povzroči posa-



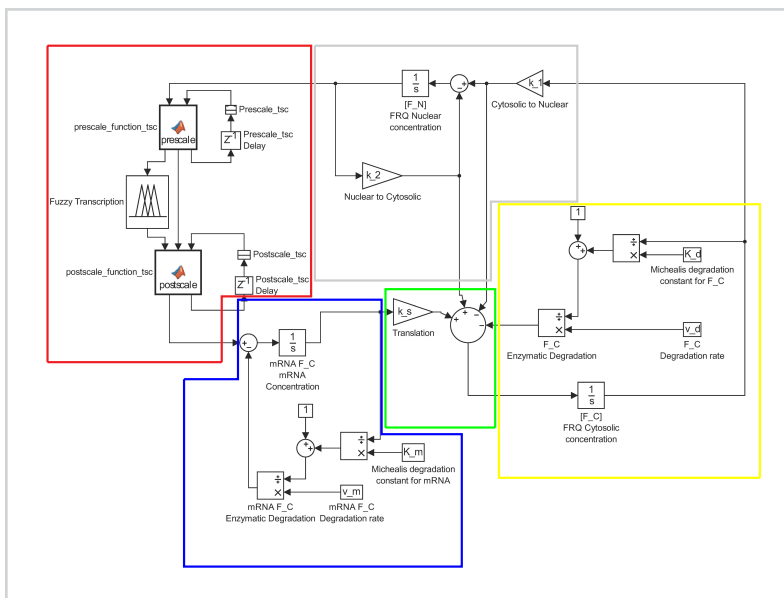
Slika 7.15

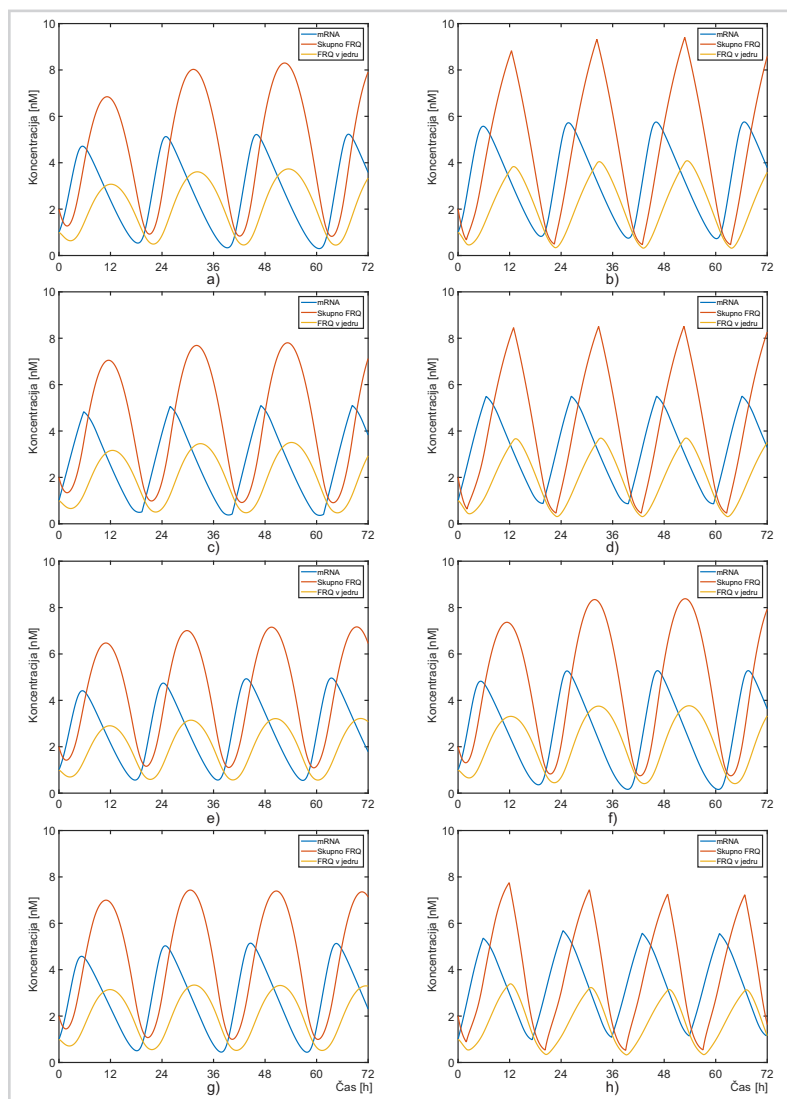
Model glive *Neurospora* v okolju MATLAB Simulink® [51]. Z rdečo je označen model procesa transkripcije, z modro procesa degradacije mRNA, z zeleno procesa translacije, z rumeno procesa degradacije proteina FRQ in s sivo prehod proteina FRQ iz jedra v citosol in iz citosola v jedro.

mezen proces. Vrednosti faktorjev se tekom simulacije avtomatično spreminjajo glede na vhodno vrednost mehke prožilne funkcije. Če več procesov modeliramo z mehкими prožilnimi funkcijami, se faktorji ne spremenijo dovolj, da bi oscilacije opisali enako kot pri izhodiščnem modelu, kar je najbolj opazno pri primeru na sliki 7.17h, pri katerem smo vse procese opisali mehko. V primeru, da maksimalne koncentracije posameznih kemijskih vrst ne ocenimo pravilno in posledično naredimo veliko napako pri nastavljanju začetnih vrednosti faktorjev za preslikavo pri funkcijah *prescale* in *postscale*, se temu primerno napaka prenese na frekvenco ter amplitudo, ki se bistveno spremenita. Kljub temu pa lahko že ob približnem poznavanju maksimalnih koncentracij in sprememb koncentracij z zamenjavo ene ali dveh prožilnih funkcij z mehкими prožilnimi funkcijami še vedno dosežemo kvantitativno relevantne simulacijske rezultate cirkadianega ritma glive *Neurospora*.

Slika 7.16

Model glive *Neurospora* v okolju MATLAB Simulink® [51], v katerem smo model transkripcije na podlagi diferencialne enačbe zamenjali z mehkim opisom. Z rdečo je označen model procesa transkripcije, z modro procesa degradacije mRNA, z zeleno procesa translacije, z rumeno procesa degradacije proteina FRQ in s sivo prehod proteina FRQ iz jedra v citosol in iz citosola v jedro.





Slika 7.17

Simulacijski rezultati modeliranja glive *Neurospora* modelov z različnimi implementacijami posameznih procesov. Izhodiščni model, ki temelji na sistemu diferencialnih enačb, je prikazan na sliki (a). Ostali simulacijski rezultati izhajajo iz modelov v katerih smo navadne prožilne funkcije enega ali več procesov zamenjali z mehкими: model z mehkim opisom translacije (b), model z mehkim opisom transkripcije (c), model z mehkim opisom translacije in transkripcije (d), model z mehkim opisom degradacije proteina FC (e), model z mehkim opisom degradacije mRNA (f), model z mehkim opisom degradacije obeh kemijskih vrst (g) in model z mehkim opisom vseh nastopajočih procesov (h).



Zaključek

V doktorski disertaciji smo predstavili dva lastna prispevka k znanosti na področju modeliranja enostavnih bioloških preklonnih sistemov. Razvili smo novo metodo za modeliranje gensko regulatornih omrežij, ki temelji na mehki logiki in ki za pridobitev simulacijskih rezultatov ne potrebuje natančnih vrednosti kinetičnih parametrov. Mehko logiko smo vpeljali v Petrijeve mreže in s tem odprli nove možnosti njihove uporabe za izgradnjo in analizo bioloških modelov.

8.1 Glavna prispevka k znanosti

Cilje obeh prispevkov smo opisali v poglavju 1. V naslednjih dveh alinejah izpostavimo izpolnitve posameznih ciljev obeh prispevkov:

- Na podlagi pregleda literature smo v poglavju 3 identificirali pomanjkljivosti obstoječih metod. Za pridobivanje kvantitativnih simulacijskih rezultatov slednje potrebujejo natančne vrednosti kinetičnih parametrov, ki pogojujejo dinamiko sistema. Natančne vrednosti slednjih pogosto niso znane, imamo pa na voljo veliko opisnega izkustvenega poznavanja dinamike posameznih procesov v bioloških sistemih, ki z uporabo obstoječih metod za deterministično modeliranje ni izkoriščeno. Osnove mehke logike in uporabo slednje za modeliranje bioloških sistemov smo predstavili v poglavjih 4 in 5. V poglavju 6 smo predstavili novo metodo za kvantitativno modeliranje, ki temelji na mehki logiki. Slednja nam omogoča, da zgradimo model na podlagi izkustvenega opisnega poznavanja procesa, s katerim je pridobitev kvantitativno relevantnih simulacijskih rezultatov možna tudi v primeru, ko natančnih vrednosti parametrov, ki pogojujejo dinamiko, ne poznamo. Na podlagi identifikacije razpoložljivih podatkov in analize vpliva parametrov na dinamiko sistema definiramo mehke spremenljivke, mehke množice, s katerimi spremenljivke predstavimo, pripadnostne funkcije, ki opisujejo mehke množice, in mehka IF-THEN pravila. S predlaganim pristopom zgradimo modele posameznih bioloških procesov [101, 102], ki nastopajo v gensko regulatornih omrežjih, in jih nato uporabimo pri izgradnji večjega modela hipotetičnega biološkega oscilatorja [74].
- Za modeliranje gensko regulatornih omrežij se kot modelirno orodje pogosto uporabljajo Petrijeve mreže. Poleg grafične predstavitve modela z grafom Petrijeve mreže nam z uporabo njihovih razširjenih definicij, ki temeljijo na obstoječih metodah za kvantitativno modeliranje, omogočajo tudi pridobivanje natančnih

simulacijskih rezultatov. Definicijo zveznih Petrijevih mrež dodatno razširimo z mehko logiko v poglavju 7. Razširitev nam omogoča uporabo mehke logike za modeliranje procesov, pri katerih natančne vrednosti parametrov niso znane, in uporabo obstoječih metod v preostalih delih. Zvezne Petrijeve mreže z mehko logiko uporabimo za izgradnjo modela cirkadianega ritma organizma glive *Neurospora*, s katerim analiziramo vpliv uporabe mehke logike pri modeliranju posameznih procesov na dinamiko gensko regulatornega omrežja. Pokažemo, da je z uporabo mehke logike napaka v simulacijskih rezultatih bistvena le v primeru, ko za vse procese v sistemu uporabimo zgolj opisno izkustveno poznavanje njihove dinamike.

8.2 Nadaljnje delo

Pričujoča doktorska disertacija ponuja več izhodišč za nadaljnje raziskovalno delo.

Pri pregledu literature smo identificirali več načinov uporabe mehke logike za kvalitativno in kvantitativno modeliranje bioloških sistemov. Pristopi uporabljajo različno definirane mehke opise za isti biološki proces. Za področje modeliranja z mehko logiko bi bilo zato smiselno vzpostaviti enotni slovar lingvističnih izrazov za opis bioloških procesov, ki bi služil kot izhodišče za izgradnjo modelov na podlagi mehke logike.

Pojavlja se tudi vprašanje uporabe mehke logike za stohastično modeliranje, pri katerem smo ravno tako kot pri determinističnih pristopih omejeni s poznavanjem natančnih vrednosti kinetičnih parametrov, ki pogojujejo dinamiko sistema. Mehko logiko bi lahko uporabili za določanje spremenljivega časovnega koraka oziroma pri izbiri reakcije, ki se v danem koraku izvede. Izognili bi se lahko tudi uporabi naključno generiranih števil, stohastičnost kemijskih reakcij pa bi lahko modelirali s spreminjajočimi lastnostmi mehkih spremenljivk, ki bi določale velikost časovnega koraka in izbiro reakcije.

V pričujoči disertaciji smo s predlaganim pristopom opisali procese gensko regulatornega omrežja, zato ostajajo odprte možnosti za uporabo mehke logike za modeliranje metabolnih omrežij in signalnih poti, pri katerih se mehka logika z obstoječimi pristopi zaenkrat uporablja le za kvalitativni opis procesov.



LITERATURA

- [1] Uri Alon. *An Introduction to Systems Biology-Design Principles of Biological Circuits*. Chapman and Hall/CRC Mathematical & Computational Biology series. Chapman & Hall/CRC, 2007. ISBN 1584886420. doi: [citeulike-article-id:1314150](https://doi.org/10.1016/j.febflet.2012.06.002). URL <http://www.loc.gov/catdir/enhancements/fy0654/2005056902-d.html>.
- [2] R Kitney and P Freemont. Synthetic biology - the state of play. *FEBS Letters*, 586(15): 2029–2036, jul 2012. ISSN 0014-5793. doi: [10.1016/j.febflet.2012.06.002](https://doi.org/10.1016/j.febflet.2012.06.002). URL <http://pubget.com/paper/22704968>.
- [3] Ernesto Andrianantoandro, Subhayu Basu, David K Karig, and Ron Weiss. Synthetic biology: New engineering rules for an emerging discipline. *Molecular Systems Biology*, May:1–14, 2006.
- [4] Timothy S Gardner, Charles R Cantor, and James J Collins. Construction of a genetic toggle switch in *Escherichia coli*. *Nature*, 403:339–342, 2000.
- [5] Jesse Stricker, Scott Cookson, Matthew R Bennett, William H Mather, Lev S Tsimring, and Jeff Hasty. A fast, robust and tunable synthetic gene oscillator. *Nature*, 456(7221):516–520, 2008. ISSN 0028-0836. doi: [10.1038/nature07389](https://doi.org/10.1038/nature07389).
- [6] Eileen Fung, Wilson W Wong, Jason K Suen, Thomas Bulter, Sun-Gu Lee, and James C Liao. A synthetic gene – metabolic oscillator. *Nature*, 435(May): 118–122, 2005. doi: [10.1038/nature03503](https://doi.org/10.1038/nature03503).
- [7] Michael B Elowitz and Stanislas Leibler. A synthetic oscillatory network of transcriptional regulators. *Nature*, 403(6767):335–338, 2000. ISSN 0028-0836. doi: [10.1038/35002125](https://doi.org/10.1038/35002125).
- [8] Natalia Strelkowa and Mauricio Barahona. Switchable genetic oscillator operating in quasi-stable mode. *Journal of The Royal Society Interface*, 7(48):1071–1082, 2010. ISSN 1742-5689. doi: [10.1098/rsif.2009.0487](https://doi.org/10.1098/rsif.2009.0487).
- [9] Yvonne Y Chen, Kate E Galloway, and Christina D Smolke. Synthetic biology: Advancing biological frontiers by building synthetic systems. *Genome biology*, 13(2):240, 2012. ISSN 1465-6906. doi: [10.1186/gb-2012-13-2-240](https://doi.org/10.1186/gb-2012-13-2-240).
- [10] Luonan Chen and Ruiqi Wang. Designing Gene Regulatory Networks With Specified Functions. *IEEE Transactions on Circuits and Systems*, 53:2444–2450, 2006.
- [11] Hidde De Jong. Modeling and Simulation of Genetic Regulatory Systems: A Literature Review. *Institut National de Recherche en Informatique et en Automatique*, 9(September):67–103, 2002. ISSN 1066-5277, 1557-8666. doi: [10.1089/10665270252833208](https://doi.org/10.1089/10665270252833208).
- [12] James N Weiss. The Hill equation revisited: Uses and misuses. *FASEB Journal*, 11:835–841, 1997.
- [13] Daniel T DT Gillespie. Stochastic simulation of chemical kinetics. *Annu. Rev. Phys. Chem.*, 58: 35–55, 2007. ISSN 0066-426X. doi: [10.1146/annurev.physchem.58.032806.104637](https://doi.org/10.1146/annurev.physchem.58.032806.104637). URL <http://www.ncbi.nlm.nih.gov/pubmed/17037977>.
- [14] Allen Cheng and Timothy K Lu. Synthetic Biology: An Emerging Engineering Discipline. *Annu. Rev. Biomed. Eng.*, 14:155–178, 2012.
- [15] Gabriele Lillacci and Mustafa Khammash. Parameter estimation and model selection in computational biology. *PLoS Computational Biology*, 6(3):e1000696, 2010. doi: [10.1371/journal.pcbi.1000696](https://doi.org/10.1371/journal.pcbi.1000696). URL <http://dx.doi.org/10.1371/journal.pcbi.1000696>.
- [16] Robert Küffner, Tobias Petri, Lukas Windhager, and Ralf Zimmer. Petri nets with fuzzy logic (PNFL): Reverse engineering and parametrization. *PLoS ONE*, 5(9):1–10, 2010. ISSN 1932-2620. doi: [10.1371/journal.pone.0012807](https://doi.org/10.1371/journal.pone.0012807).
- [17] Lukas Windhager and Ralf Zimmer. Intuitive modeling of dynamic systems with Petri nets and fuzzy logic. In *Proc German Conf Bioinf*, volume 136, pages 106–115. Citeseer, 2008.

- [18] Yves Gendraul, Morgan Madec, Vincent Wlotzko, Christophe Lalletment, and Jacques Haiech. Fuzzy logic, an intermediate description level for design and simulation in synthetic biology. *2013 IEEE Biomedical Circuits and Systems Conference, BioCAS 2013*, pages 370–373, 2013. ISSN 2163-4025. doi: [10.1109/BioCAS.2013.6679716](https://doi.org/10.1109/BioCAS.2013.6679716).
- [19] RaedI. Hamed, SyedI. Ahson, and Rafat Parveen. Designing genetic regulatory networks using fuzzy Petri nets approach. *International Journal of Automation and Computing*, 7(3):403–412, 2010. ISSN 1476-8186. doi: [10.1007/s11633-010-0521-9](https://doi.org/10.1007/s11633-010-0521-9). URL <http://dx.doi.org/10.1007/s11633-010-0521-9>.
- [20] Lukas Windhager. *Modeling of dynamic systems with Petri nets and fuzzy logic*. PhD thesis, Ludwig-Maximilians-Universität München, apr 2013. URL <http://nbn-resolving.de/urn:nbn:de:bvb:19-156555>.
- [21] Monika Heiner, David Gilbert, and Robin Donaldson. Petri nets for systems and synthetic biology. *Formal methods for Computational Systems Biology*, pages 215–264, 2008.
- [22] Allison P Heath and Lydia E Kavrakli. Computational challenges in systems biology. *Computer Science Review*, 3(1):1–17, 2009.
- [23] Christina M Agapakis and Pamela A Silver. Synthetic biology: exploring and exploiting genetic modularity through the design of novel biological networks. *Molecular Biosystems*, 5:704–713, 2009.
- [24] Mariano Bizzarri, Alissandra Cucina, and Alessandro Palombo. *The Conceptual Foundations of Systems Biology: An Introduction*. Nova Science Publishers, Inc., 2014. ISBN 9781631172557.
- [25] Ira Mellman and Tom Misteli. Computational cell biology. *Journal of Cell Biology*, 161(3):463–464, 2003. ISSN 00219525. doi: [10.1083/jcb.200303202](https://doi.org/10.1083/jcb.200303202).
- [26] Eric Davidson and Michael Levin. Gene regulatory networks. *Proceedings of the National Academy of Sciences of the United States of America*, 102(14):4935, 2005. ISSN 00278424. doi: [10.1073/pnas.0502024102](https://doi.org/10.1073/pnas.0502024102). URL [http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=556010\(&\)tool=pmcentrez\(&\)rendertype=abstract](http://www.pubmedcentral.nih.gov/articlerender.fcgi?artid=556010(&)tool=pmcentrez(&)rendertype=abstract).
- [27] Guy Karlebach and Ron Shamir. Modelling and analysis of gene regulatory networks. *Nat Rev Mol Cell Biol*, 9(10):770–780, oct 2008. ISSN 1471-0080. doi: [10.1038/nrm2503](https://doi.org/10.1038/nrm2503). URL <http://www.ncbi.nlm.nih.gov/pubmed/18797474><http://dx.doi.org/10.1038/nrm2503>.
- [28] Thomas Schlitt and Alvis Brazma. Current approaches to gene regulatory network modelling. *BMC Bioinformatics*, 8(Suppl 6):S9, 2007. ISSN 1471-2105. doi: [10.1186/1471-2105-8-S6-S9](https://doi.org/10.1186/1471-2105-8-S6-S9). URL <http://www.biomedcentral.com/1471-2105/8/S6/S9>.
- [29] Thomas Schlitt and Alvis Brazma. Learning about Gene Regulatory Networks from Gene Deletion Experiments. *Comparative and Functional Genomics*, 3(6):499–503, 2002. ISSN 1531-6912. doi: [10.1002/cfg.220](https://doi.org/10.1002/cfg.220). URL <http://www.hindawi.com/journals/ijg/2002/459080/abs/>.
- [30] Leon Glass and Stuart A. Kauffman. The logical analysis of continuous, non-linear biochemical control networks. *Journal of Theoretical Biology*, 39(1):103–129, apr 1973. ISSN 00225193. doi: [10.1016/0022-5193\(73\)90208-7](https://doi.org/10.1016/0022-5193(73)90208-7). URL <http://linkinghub.elsevier.com/retrieve/pii/S0022519373902087>.
- [31] Richard Banks and L Jason Steggles. A High-Level Petri Net Framework for Genetic Regulatory Networks. Technical Report 3, School of Computing Science, University of Newcastle, 2007.
- [32] Ilya Shmulevich, Edward R. Dougherty, Seungchan Kim, and Wei Zhang. Probabilistic Boolean networks: A rule-based uncertainty model for gene regulatory networks. *Bioinformatics*, 18(2)(2):261–274, feb 2002. ISSN 1367-4803. doi: [10.1093/bioinformatics/18.2.261](https://doi.org/10.1093/bioinformatics/18.2.261). URL <http://bioinformatics.oxfordjournals.org/cgi/doi/10.1093/bioinformatics/18.2.261>.
- [33] Ilya Shmulevich, Ilya Gluhovsky, Ronaldo F. Hashimoto, Edward R. Dougherty, and Wei Zhang. Steady-State Analysis of Genetic Regulatory Networks Modelled by Probabilistic Boolean Networks. *Comparative and Functional Genomics*, 4(6):601–608, 2003. ISSN 1531-6912. doi: [10.1002/cfg.342](https://doi.org/10.1002/cfg.342). URL <http://www.hindawi.com/journals/ijg/2003/131563/abs/>.
- [34] Ina Koch, Markus Schüller, and Monika Heiner. STEPP - Search Tool for Exploration of Petri net Paths: A New Tool for Petri Net-Based Path Analysis in Biochemical Networks. *Studies in health technology and informatics*, 162: 113–21, 2011. ISSN 0926-9630. URL <http://www.ncbi.nlm.nih.gov/pubmed/21685567>.
- [35] Judea Pearl. *Probabilistic Reasoning in Intelligent Systems: Networks of Plausible Inference*. Morgan Kaufmann, 1988.
- [36] Nir Friedman, Michal Linial, Iftach Nachman, and Dana Pe'er. Using Bayesian Networks to Analyze Expression Data. *Journal of Computational Biology*, 7(3-4):601–620, aug 1999. ISSN 1066-5277. doi: [10.1089/106652700750050961](https://doi.org/10.1089/106652700750050961). URL <http://www.liebertonline.com/doi/abs/10.1089/106652700750050961>.

- [37] Eran Segal, Michael Shapira, Aviv Regev, Dana Pe'er, David Botstein, Daphne Koller, and Nir Friedman. Module networks: Identifying regulatory modules and their condition-specific regulators from gene expression data. *Nature Genetics*, 34(2):166–176, jun 2003. doi: [10.1038/ng1165](https://doi.org/10.1038/ng1165). URL <http://www.nature.com/doi/finder/10.1038/ng1165>.
- [38] Nir Friedman. Inferring cellular networks using probabilistic graphical models. *Science (New York, N.Y.)*, 303(5659):799–805, feb 2004. ISSN 1095-9203. doi: [10.1126/science.1094068](https://doi.org/10.1126/science.1094068). URL <http://www.ncbi.nlm.nih.gov/pubmed/14764868>.
- [39] James Keener and James Sneyd. *Mathematical Physiology*, volume 8/1 of *Interdisciplinary Applied Mathematics*. Springer New York, New York, NY, 2009. ISBN 978-0-387-75846-6. doi: [10.1007/978-0-387-75847-3](https://doi.org/10.1007/978-0-387-75847-3). URL <http://link.springer.com/10.1007/978-0-387-75847-3>.
- [40] John McMurry and Robert C Fay. *Chemistry, 4th edition*. Prentice Hall, 2007.
- [41] David L Nelson and Michael M Cox. *Lehninger Principles of Biochemistry, Fourth Edition*. W. H. Freeman, 2004.
- [42] James D Murray. *Mathematical biology I: An introduction, Vol. 17 of interdisciplinary applied mathematics*. Springer, New York, NY, USA, 2002.
- [43] Von L Michaelis, Miss L Maud Menten, Roger S Goody, and Kenneth A Johnson. Die Kinetik der Invertinwirkung. Technical report, 1913.
- [44] D T Gillespie. *Markov Processes: An Introduction for Physical Scientists*. Academic Press: San Diego, 1992.
- [45] Raya Khanin and Desmond J Higham. Chemical Master Equation and Langevin Regimes for a Gene Transcription Model. *Theoretical Computer Science*, 408:31–40, 2008.
- [46] Verena Wolf, Goel Rushil, Maria Mateescu, and Thomas A Henzinger. Solving the chemical master equation using sliding windows. *BMC Systems Biology*, 4:1–19, 2010.
- [47] Daniel T Gillespie. A General Method for Numerically Simulating the Stochastic Time Evolution of Coupled Chemical Reactions. *Journal of Computational Physics*, 22:403–434, 1976.
- [48] Daniel T Gillespie. Approximate accelerated stochastic simulation of chemically reacting systems. *Journal of Chemical Physics*, 115:1716–1733, 2001.
- [49] Hana El Samad, Mustafa Khammash, Linda Petzold, and Dan Gillespie. Stochastic Modeling of Gene Regulatory Networks. *International Journal of Robust and Nonlinear Control*, 15:691–711, 2005.
- [50] Hans-Jürgen Zimmermann. *Fuzzy set theory and its applications*. Springer Science & Business Media, USA, 2001.
- [51] MATLAB. *version 8.50 (R2015a)*. The MathWorks Inc., Natick, Massachusetts, 2015.
- [52] George J Klir and Bo Yuan. *Fuzzy sets and fuzzy logic*. Prentice Hall New Jersey, 1995.
- [53] J Virant. *Čas v mehkih sistemih*. Didakta, Slovenija, 1998. ISBN 9789616214391.
- [54] Aris Tepeli and Amable Hortaçu. A fuzzy logic approach for regulation in flux balance analysis. *Biochemical Engineering Journal*, 39(1):137–148, 2008. ISSN 1369703X. doi: [10.1016/j.bej.2007.08.022](https://doi.org/10.1016/j.bej.2007.08.022).
- [55] Bree B. Aldridge, Julio Saez-Rodriguez, Jeremy L. Muhlich, Peter K. Sorger, and Douglas A. Lauffenburger. Fuzzy logic analysis of kinase pathway crosstalk in TNF/EGF/insulin-induced signaling. *PLoS Computational Biology*, 5(4):e1000340, 2009. ISSN 1553734X. doi: [10.1371/journal.pcbi.1000340](https://doi.org/10.1371/journal.pcbi.1000340).
- [56] I A Maraziotis, A Dragomir, and A Bezerianos. Gene networks reconstruction and time-series prediction from microarray data using recurrent neural fuzzy networks. *IET Systems Biology*, 1(1):41–50, 2007.
- [57] Zuyi Huang and Juergen Hahn. Fuzzy modeling of signal transduction networks. *Chemical Engineering Science*, 64(9):2044–2056, 2009. ISSN 00092509. doi: [10.1016/j.ces.2009.01.041](https://doi.org/10.1016/j.ces.2009.01.041). URL <http://dx.doi.org/10.1016/j.ces.2009.01.041>.
- [58] C Munoz, F Vargas, J Bustos, C Muñoz, F Vargas, J Bustos, M Curilem, S Salvo, and H Miranda. Fuzzy logic in genetic regulatory network models. *Int. J. of Computers, Communications & Control*, IV(4): 363–373, 2009.
- [59] Peter J Woolf and Yixin Wang. A fuzzy logic approach to analyzing gene expression data. *Physiological Genomics*, 3(1):9–15, 2000. ISSN 1531-2267.
- [60] R. Ram, M. Chetty, and T.I. Dix. Fuzzy Model for Gene Regulatory Network. *2006 IEEE International Conference on Evolutionary Computation*, pages 1450–1455, 2006. doi: [10.1109/CEC.2006.1688479](https://doi.org/10.1109/CEC.2006.1688479). URL <http://ieeexplore.ieee.org/lpdocs/epic03/wrapper.htm?arnumber=1688479>.
- [61] Guy N Brock, Vasyil Pihur, and Laura Kubatko. Detecting Gene Regulatory Networks from Microarray Data Using Fuzzy Logic. *Fuzzy Systems in Bioinformatics and Computational Biology*, pages 141–163, 2009.

- [62] Wolfgang Schmidt-Heck, Madlen Matz-Soja, Susanne Aleithe, Eugenia Marbach, Reinhard Guthke, and Rolf Gebhardt. Fuzzy modeling reveals a dynamic self-sustaining network of the GLI transcription factors controlling important metabolic regulators in adult mouse hepatocytes. *Molecular bioSystems*, 11(8):2190–2197, 2015. ISSN 1742-2051. doi: [10.1039/c5mb00129c](https://doi.org/10.1039/c5mb00129c). URL <http://pubs.rsc.org/en/content/articlehtml/2015/mb/c5mb00129c>.
- [63] Habtom Resson, Robert Reynolds, and Rency S Varghese. Increasing the efficiency of fuzzy logic-based gene expression data analysis. *Physiological genomics*, 13(2):107–117, 2003. ISSN 1531-2267. doi: [10.1152/physiolgenomics.00097.2002](https://doi.org/10.1152/physiolgenomics.00097.2002).
- [64] Bahrad A Sokhansanj, J Patrick Fitch, Judy N Quong, and Andrew A Quong. Linear fuzzy gene network models obtained from microarray data by exhaustive search. *BMC Bioinformatics*, 5(1):1, 2004. ISSN 1471-2105. doi: [10.1186/1471-2105-5-108](https://doi.org/10.1186/1471-2105-5-108).
- [65] Pan Du, Jian Gong, Eve Syrkin Wurtele, Julie A Dickerson, Others, Eve Syrkin Wurtele, Julie A Dickerson, Others, Eve Syrkin Wurtele, Julie A Dickerson, Eve Syrkin Wurtele, and Julie A Dickerson. Modeling gene expression networks using fuzzy logic. *IEEE Transactions on Systems, Man, and Cybernetics, Part B: Cybernetics*, 35(6):1351–1359, 2005. ISSN 1083-4419. doi: [10.1109/TSMCB.2005.855590](https://doi.org/10.1109/TSMCB.2005.855590).
- [66] Patrick C H Ma and Keith C C Chan. Inferring gene regulatory networks from expression data by discovering fuzzy dependency relationships. *IEEE Transactions on Fuzzy Systems*, 16(2):455–465, 2008.
- [67] Xiaohua Hu, Bahrad Sokhansanj, Daniel Wu, and Yuchun Tang. A novel approach for mining and fuzzy simulation of subnetworks from large biomolecular networks. *IEEE Transactions on Fuzzy Systems*, 15(6):1219–1229, 2007. ISSN 10636706. doi: [10.1109/TFUZZ.2007.896248](https://doi.org/10.1109/TFUZZ.2007.896248).
- [68] Raed I Hamed, S I Ahsorr, R Parveerr, S I Ahson, and Rafat Parveen. From fuzzy logic theory to fuzzy Petri nets predicting changes in gene expression level. In *Proceeding of International Conference on Methods and Models in Computer Science, 2009.*, pages 1–7. IEEE, 2009.
- [69] R I Hamed. Intelligent method of Petri net formal computational modeling of biological networks. In *Computer Science and Electronic Engineering Conference (CEECE)*, pages 162–167, sep 2013. doi: [10.1109/CEECE.2013.6659465](https://doi.org/10.1109/CEECE.2013.6659465).
- [70] Gang Sun, Yonghuiand Feng and Jinde Cao. A New Approach to Dynamic Fuzzy Modeling of Genetic Regulatory Networks. *IEEE Transactions on Nano-Bioscience*, 9(4):263–272, 2010. ISSN 1536-1241. doi: [10.1109/TNB.2010.2082559](https://doi.org/10.1109/TNB.2010.2082559).
- [71] Melody K. Morris, Julio Saez-Rodriguez, David C. Clarke, Peter K. Sorger, and Douglas A. Lauffenburger. Training signaling pathway maps to biochemical data with constrained fuzzy logic: Quantitative analysis of liver cell responses to inflammatory stimuli. *PLoS Computational Biology*, 7(3):e1001099–e1001099, 2011. ISSN 1553734X. doi: [10.1371/journal.pcbi.1001099](https://doi.org/10.1371/journal.pcbi.1001099).
- [72] Ricardo Linden and Amit Bhaya. Evolving fuzzy rules to model gene expression. *BioSystems*, 88(1-2):76–91, 2007. ISSN 03032647. doi: [10.1016/j.biosystems.2006.04.006](https://doi.org/10.1016/j.biosystems.2006.04.006).
- [73] Edda Klipp and et al. *Systems Biology in Practice*. John Wiley & Sons, 2008, 2007. ISBN 9783527310784. doi: [10.1002/3527603603](https://doi.org/10.1002/3527603603). URL papers2://publication/uuid/1D95D849-C0E8-4CBB-9686-0D1396EE8BC9.
- [74] Jure Bordon, Miha Moškon, Nikolaj Zimic, and Miha Mraz. Fuzzy logic as a computational tool for quantitative modelling of biological systems with uncertain kinetic data. *IEEE/ACM Transactions on Computational Biology and Bioinformatics*, 12(5):1199–1205, 2015. ISSN 15455963. doi: [10.1109/TCBB.2015.2424242](https://doi.org/10.1109/TCBB.2015.2424242).
- [75] Paolo Prandoni and Martin Vetterli. *Signal processing for Communications*. Taylor & Francis Group, 2008. ISBN 9782940222029. doi: [10.1007/SpringerReference_24796](https://doi.org/10.1007/SpringerReference_24796).
- [76] Jure Bordon, Miha Moškon, Nikolaj Zimic, and Miha Mraz. Semi-quantitative modelling of gene regulatory processes with unknown parameter values using Fuzzy logic and Petri nets. *V reviziji Fundamenta Informaticae*, 2016.
- [77] Tadao Murata. Petri nets: Properties, analysis and applications. *Proceedings of the IEEE*, 77(4):541–580, 1989. ISSN 00189219. doi: [10.1109/5.24143](https://doi.org/10.1109/5.24143). URL <http://ieeexplore.ieee.org/lpdocs/epic83/wrapper.htm?arnumber=24143>.
- [78] James Lyle Peterson. *Petri Net Theory and the Modeling of Systems*. Prentice Hall PTR, Upper Saddle River, NJ, USA, 1981. ISBN 0136619835.
- [79] Carl G. Looney. Fuzzy Petri Nets for Rule-Based Decision making. *IEEE Transactions on Systems, Man and Cybernetics*, 18(1):178–183, 1988. ISSN 21682909. doi: [10.1109/21.87067](https://doi.org/10.1109/21.87067).
- [80] Shyi-Ming Chen, Jyh-sheng Ke, and Jin-Fu Chang. Knowledge representation using fuzzy Petri nets. *IEEE Transactions on Knowledge and Data Engineering*, 2(3):311–319, 1990. ISSN 10414347. doi: [10.1109/69.60794](https://doi.org/10.1109/69.60794). URL <http://ieeexplore.ieee.org/lpdocs/epic83/wrapper.htm?arnumber=68794>.

- [81] W Pedrycz and F Gomide. A generalized Fuzzy Petri net model. *IEEE Transactions on Fuzzy Systems*, 2(4): 295–301, 1994.
- [82] Samir M Koriem. A Fuzzy Petri Net Tool for Modeling and Verification of Knowledge-Based Systems. *The Computer Journal*, 43(3):206–233, 2000. ISSN 00104620. URL [http://www.scopus.com/inward/record.url?eid=2-s2.0-0033701673\(&\)partnerID=40\(&\)md5=1a4c343a62679a4dce9edb83ac8305a8](http://www.scopus.com/inward/record.url?eid=2-s2.0-0033701673(&)partnerID=40(&)md5=1a4c343a62679a4dce9edb83ac8305a8).
- [83] Raed I Hamed, S I Ahson, and R Parveen. A new approach for modelling gene regulatory networks using fuzzy petri nets. *Journal of Integrative Bioinformatics*, 7(1):1–16, 2010. ISSN 1613-4516. doi: [10.2390/bicoll-jib-2010-113](https://doi.org/10.2390/bicoll-jib-2010-113). URL <http://www.ncbi.nlm.nih.gov/pubmed/20134077>.
- [84] Hassane Alla and Rem David. A modelling and analysis tool for discrete events systems : continuous Petri net. *Performance Evaluation*, 33:175–199, 1998. ISSN 01665316. doi: [10.1016/S0166-5316\(98\)00016-9](https://doi.org/10.1016/S0166-5316(98)00016-9).
- [85] V N Reddy, M N Liebman, and M L Mavrouniotis. Qualitative analysis of biochemical reaction systems. *Computers in biology and medicine*, 26(1):9–24, 1996. ISSN 0010-4825. doi: [10.1016/0010-4825\(95\)00042-9](https://doi.org/10.1016/0010-4825(95)00042-9). URL <http://www.ncbi.nlm.nih.gov/pubmed/8654057>.
- [86] Ina Koch. Petri nets - A mathematical formalism to analyze chemical reaction networks. *Molecular Informatics*, 29(12):838–843, 2010. ISSN 18681743. doi: [10.1002/minf.201000086](https://doi.org/10.1002/minf.201000086).
- [87] Andrea Sackmann, Monika Heiner, and Ina Koch. Application of Petri net based analysis techniques to signal transduction pathways. *BMC Bioinformatics*, 7(1):482, 2006. ISSN 1471-2105. doi: [10.1186/1471-2105-7-482](https://doi.org/10.1186/1471-2105-7-482). URL <http://bmcbioinformatics.biomedcentral.com/articles/10.1186/1471-2105-7-482>.
- [88] Ina Koch, Björn H Junker, and Monika Heiner. Application of Petri net theory for modelling and validation of the sucrose breakdown pathway in the potato tuber. *Bioinformatics (Oxford, England)*, 21(7):1219–26, apr 2005. ISSN 1367-4803. doi: [10.1093/bioinformatics/bti145](https://doi.org/10.1093/bioinformatics/bti145). URL <http://www.ncbi.nlm.nih.gov/pubmed/15546934>.
- [89] H Matsuno, Atsushi Doi, Masao Nagasaki, and Satoru Miyano. Hybrid Petri net representation of gene regulatory network. *Pacific Symposium On Biocomputing*, 349(338-349):341–352, 2000. ISSN 1793-5991. doi: [10.1093/bioinformatics/bti145](https://doi.org/10.1093/bioinformatics/bti145). URL <http://www.ncbi.nlm.nih.gov/pubmed/10902182>.
- [90] Claudine Chaouiya, Elisabeth Remy, Paul Ruet, and Denis Thieffry. Qualitative modelling of genetic networks: From logical regulatory graphs to standard petri nets. *Applications and Theory of Petri Nets 2004*, pages 137–156, 2004. ISSN 03029743.
- [91] L. Jason Steggles, Richard Banks, Oliver Shaw, and Anil Wipat. Qualitatively modelling and analysing genetic regulatory networks: A Petri net approach. *Bioinformatics*, 23(3):336–343, feb 2007. ISSN 13674803. doi: [10.1093/bioinformatics/btl596](https://doi.org/10.1093/bioinformatics/btl596). URL <http://bioinformatics.oxfordjournals.org/content/23/3/336.abstract>.
- [92] Jean-Paul P Comet, Hanna Klaudel, and Stéphane Liauzu. Modeling multi-valued genetic regulatory networks using high-level Petri nets. *Applications and Theory of Petri Nets 2005*, 3536:986, 2005.
- [93] C. Chaouiya, A. Naldi, E. Remy, and D. Thieffry. Petri net representation of multi-valued logical regulatory graphs. *Natural Computing*, 10(2):727–750, 2011. ISSN 1567-7818. doi: [10.1007/s11047-010-9178-0](https://doi.org/10.1007/s11047-010-9178-0). URL <http://dx.doi.org/10.1007/s11047-010-9178-0>.
- [94] Hiroshi Matsuno, Masao Nagasaki, and Satoru Miyano. Hybrid Petri net based modeling for biological pathway simulation. *Natural Computing*, 10(3):1099–1120, sep 2011. ISSN 15677818. doi: [10.1007/s11047-009-9164-6](https://doi.org/10.1007/s11047-009-9164-6). URL <http://link.springer.com/10.1007/s11047-009-9164-6>.
- [95] Peter J E Goss and Jean Peccoud. Quantitative modeling of stochastic systems in molecular biology by using stochastic Petri nets. *PNAS*, 95(12):6750–6755, 1998. ISSN 0027-8424. doi: [10.1073/pnas.95.12.6750](https://doi.org/10.1073/pnas.95.12.6750). URL <http://www.pnas.org/cgi/content/abstract/95/12/6750>.
- [96] Oliver Shaw, Jason Steggles, and Anil Wipat. Automatic Parameterisation of Stochastic Petri Net Models of Biological Networks. *Electronic Notes in Theoretical Computer Science*, 151(3):111–129, 2006. ISSN 15710661. doi: [10.1016/j.entcs.2006.03.015](https://doi.org/10.1016/j.entcs.2006.03.015).
- [97] David Gilbert and Monika Heiner. LNCS 4024 - From Petri Nets to Differential Equations - An Integrative Approach for Biochemical Network Analysis. *Lecture Notes in Computer Science (including subseries Lecture Notes in Artificial Intelligence and Lecture Notes in Bioinformatics)*, 4024(December):181–200, 2006. ISSN 16113349. doi: [10.1007/11767589_11](https://doi.org/10.1007/11767589_11).
- [98] Fei Liu, Monika Heiner, and Ming Yang. Fuzzy Stochastic Petri Nets for Modeling Biological Systems with Uncertain Kinetic Parameters. *PLoS One*, 11(2):e0149674, 2016. ISSN 1932-6203. doi: [10.1371/journal.pone.0149674](https://doi.org/10.1371/journal.pone.0149674). URL <http://dx.plos.org/10.1371/journal.pone.0149674>.
- [99] Inho Park, Dokyun Na, Doheon Lee, and Kwang H. Lee. *Fuzzy Continuous Petri Net-Based Approach for Modeling Immune Systems*, pages 278–285. Springer

- Berlin Heidelberg, Berlin, Heidelberg, 2006. ISBN 978-3-540-33184-1. doi: [10.1007/11731177_35](https://doi.org/10.1007/11731177_35). URL http://dx.doi.org/10.1007/11731177_35.
- [100] J.-C. Leloup, D. Gonze, and A. Goldbeter. Limit Cycle Models for Circadian Rhythms Based on Transcriptional Regulation in *Drosophila* and *Neurospora*. *Journal of Biological Rhythms*, 14(6):433–448, dec 1999. ISSN 0748-7304. doi: [10.1177/074873099129000948](https://doi.org/10.1177/074873099129000948). URL <http://jbr.sagepub.com/cgi/doi/10.1177/074873099129000948>.
- [101] Jure Bordon, Miha Moškon, and Miha Mraz. Modeling gene regulatory networks using Petri Nets. *Elektrotehniški Vestnik/Electrotechnical Review*, 79(4): 189–192, 2012. ISSN 00135852 22323228.
- [102] Jure Bordon, Miha Moškon, and Miha Mraz. Overcoming unknown kinetic data for quantitative modeling of biological systems using fuzzy logic and Petri nets. In *CEUR Workshop Proceedings*, volume 1159, pages 3–14, 2014.